Thoughts on Universal Tester Board design for L2 data inputs

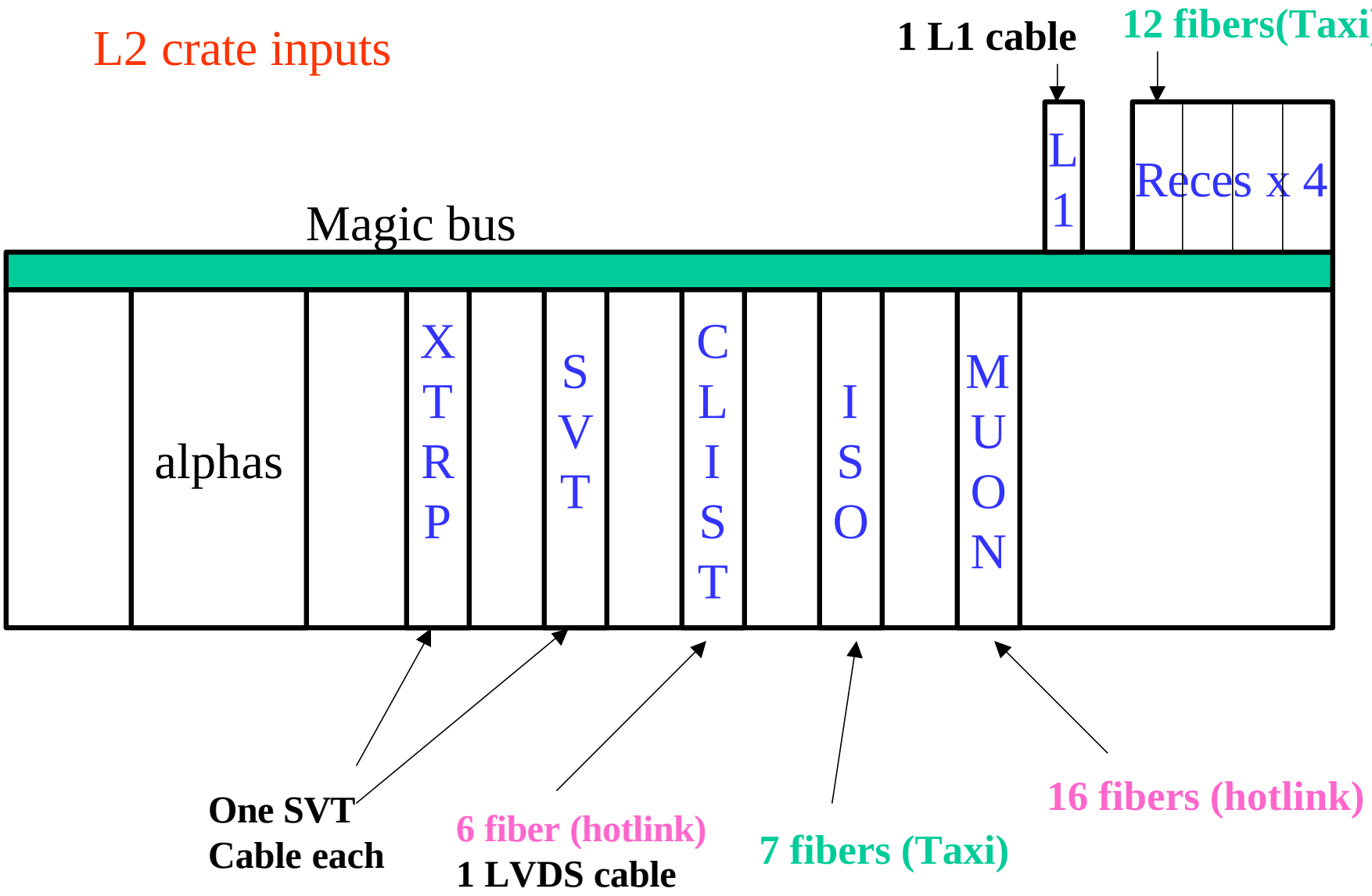
Oct. 10, 01

Ted Liu

Outline

- **L2 decision crate inputs**
- **Requirements for Universal Test Board**
- **Block diagrams, interfaces**
- **FPGA choice (example: Altera APEX20K)**
- **RAM implementation (examples)**
- **Worse case I/O pin counts**
- **how many boards needed**
- **Optional: spy backplane signals**
- **Teststand setup**
- **Misc.**

L2 crate inputs

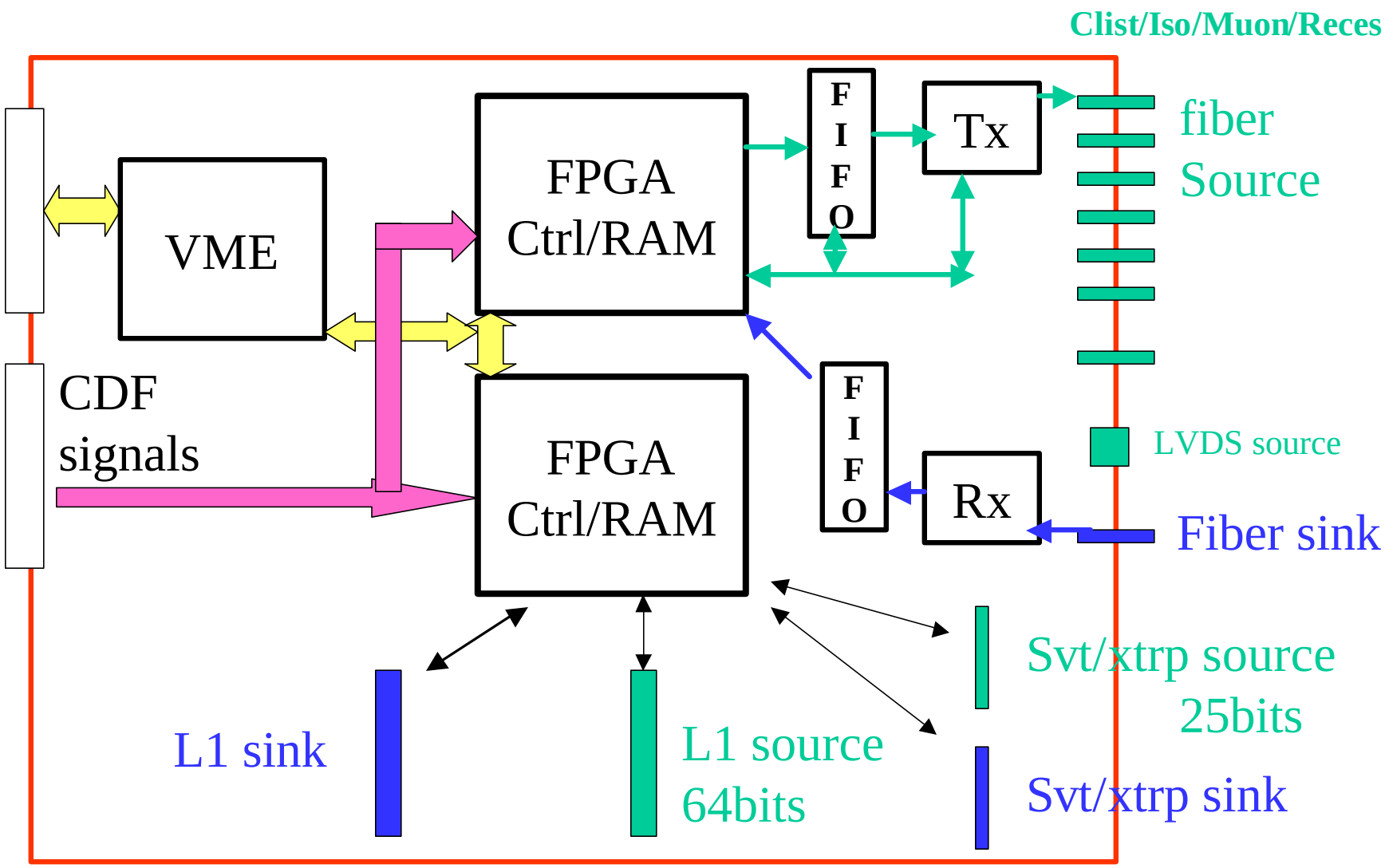


Goal: build one type of test board to provide data source for all inputs

Wish List:

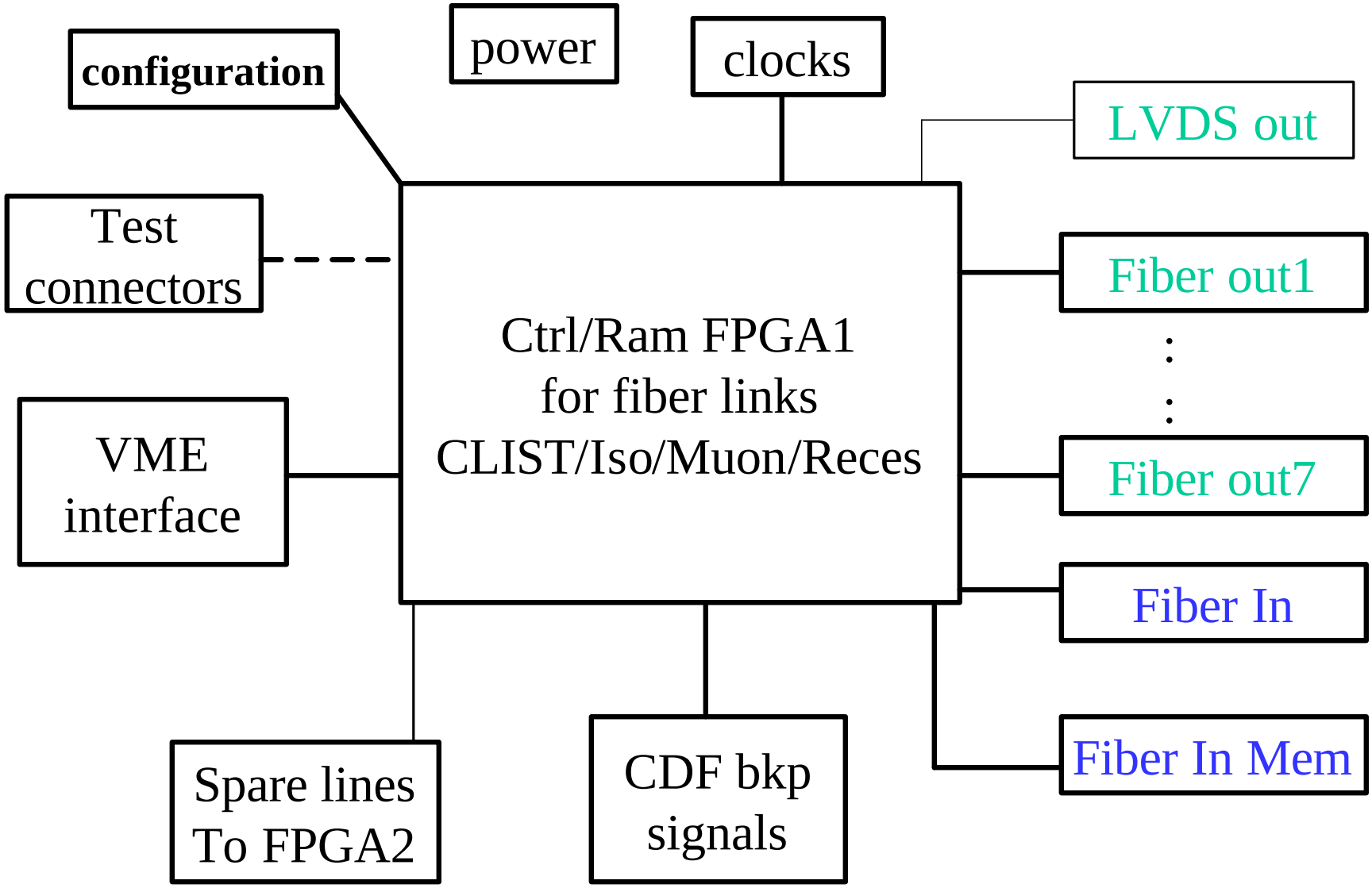
One type of board capable to provide data source for all;
VME interface: download/read memory patterns;
Aware of all CDF signals on the backplane;
Triggered on L1A+buffer#, external trigger?
Memory pattern selection with buffer numbers;
Self test, one data sink per output connector type;
Sink recording memory deep enough;
Number of output (enable) selectable via VME;
Adjustable latency for all output;
Independent clocks for each output;
HRR
…
Optional: spy on magic bus P3 and P2 signals;

Block diagram: 9U VME

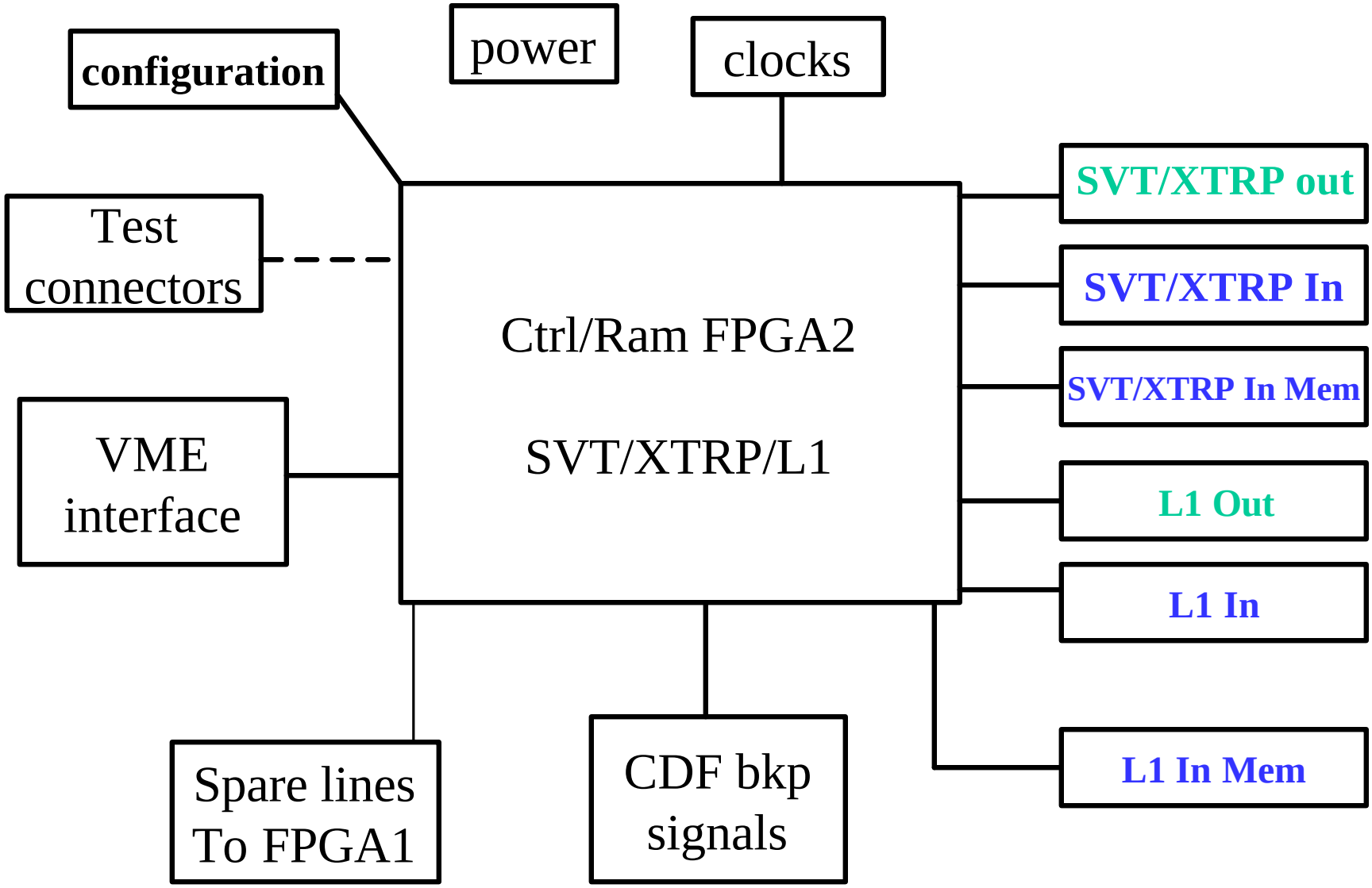


*option: spy on magic bus signals?

Ctrl/RAM FPGA1 for fiber output block diagram



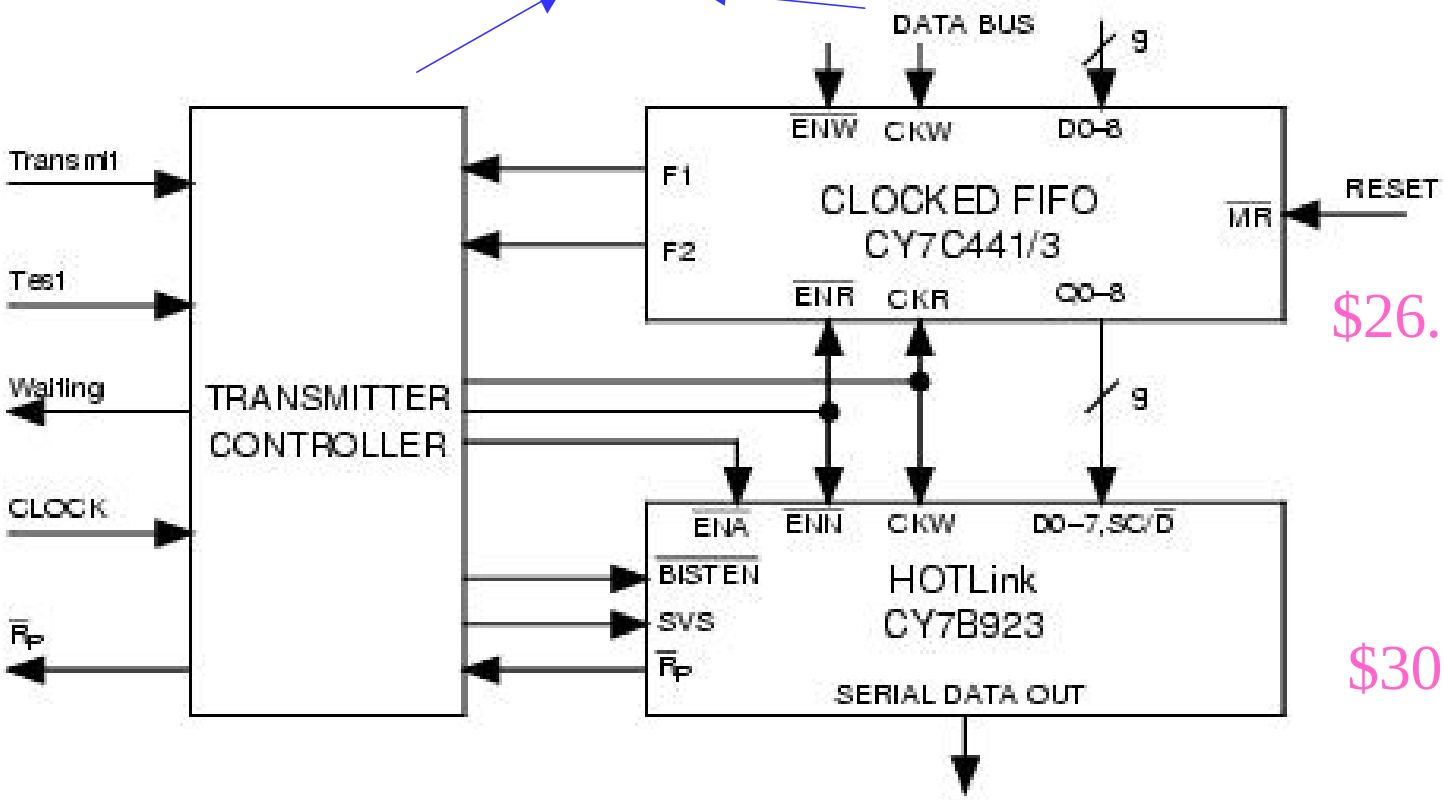
Ctrl/RAM FPGA2 for SVT/XTRP/L1 output block diagram



Fiber source (hotlink) block

each source has
its own oscillator

From Ctrl/RAM FPGA1



$26.5 x7

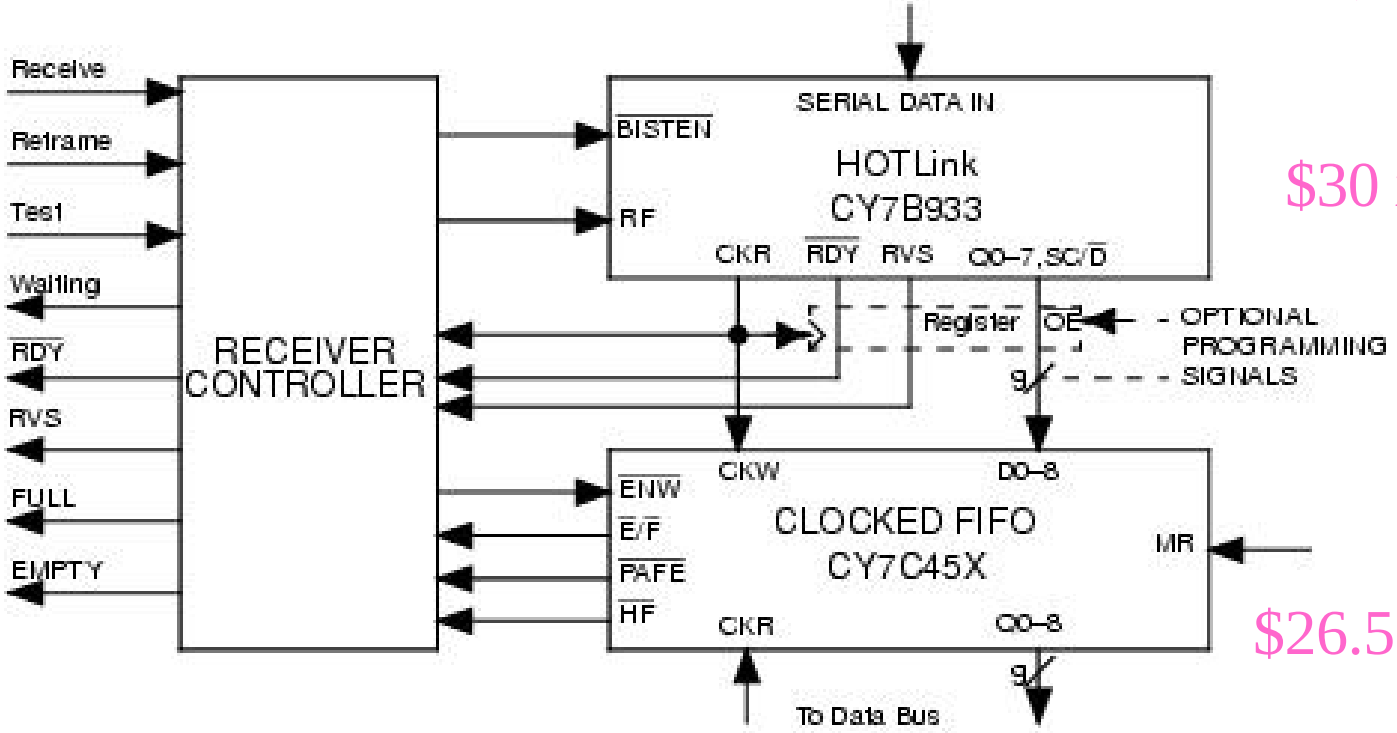
$30 x 7

Use hotlink BIST(built-In Self-Test)

Transmitter module 269051-1
need to find replacement

Fiber sink (hotlink) block

Need replacement for
Receiver module 269052 ($? x1)



$30 x1

$26.5 x1

Ctrl/RAM FPGA

Use hotlink BIST(built-In Self-Test)

mem deep

LVDS output (for CLIST) block

The signals come from CLIQUE, driven by 16 ns clock (1/8 CDFCLK). Only have a few bits information (event_done and buffer bits). The LVDS driver will use DS90c031 (a few $) as done on CLIQUE. Event_done is used to signal that all The clusters for a given event have been sent to CLIST.

May not need sink for this output, will use test pins instead.
The relative latency will be adjustable via VME.
In fact, can be controlled by memory patterns.

But having a sink is probably useful to record data from real source… for example, to check the relative timing and catch out-of-sync error … etc.

Fiber source (Taxi) block

This one is hard, as the component (AM7968) no longer exists. Find a trick on the web …

Replace Your Am7968 TAXI™ Transmitter With a CY7B923 HOTLink™

Overview

The Am7968 TAXI transmitter, when operating in 8-bit mode, uses a 4B/5B encoding scheme to convert input data and commands into a form suitable for serial transmission and clock recovery. Communication with an existing Am7969 TAXI receiver requires the use of this same encoding scheme, presented in the same form and data-rate as that generated by the Am7968. By operating the CY7B923 HOTLink Transmitter in Bypass mode (unencoded 10-bit data path) mated to a small PLD, it is possible to exactly emulate the 4B/5B encoding used by the Am7968.

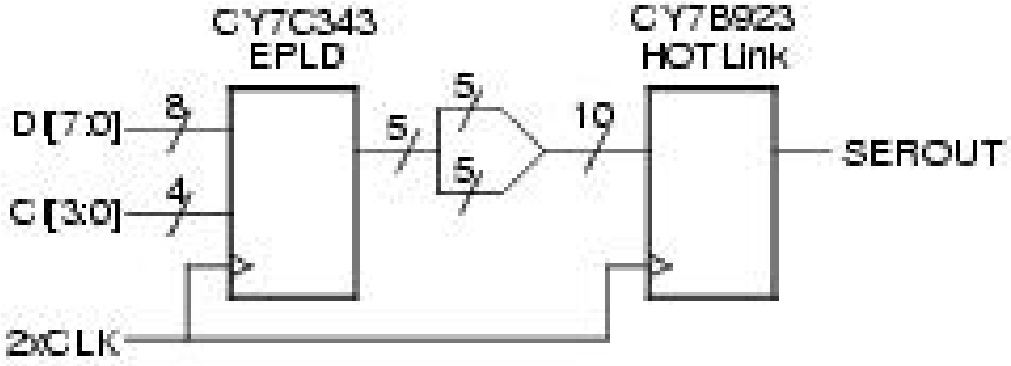


Figure 4. Am7968 Emulator Block Diagram

Figure 1-1 Am7968 TAXI Transmitter Block Diagram

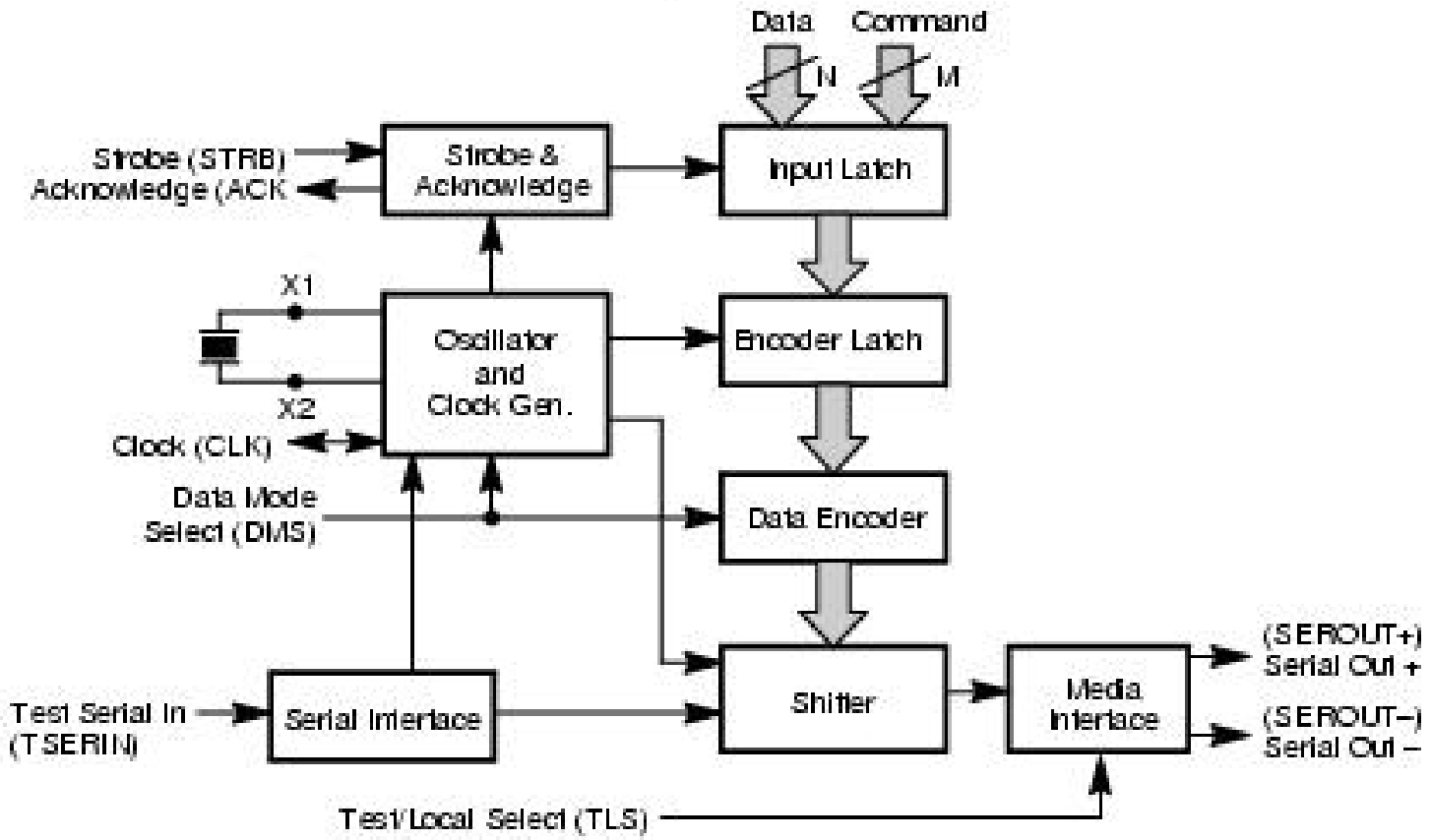
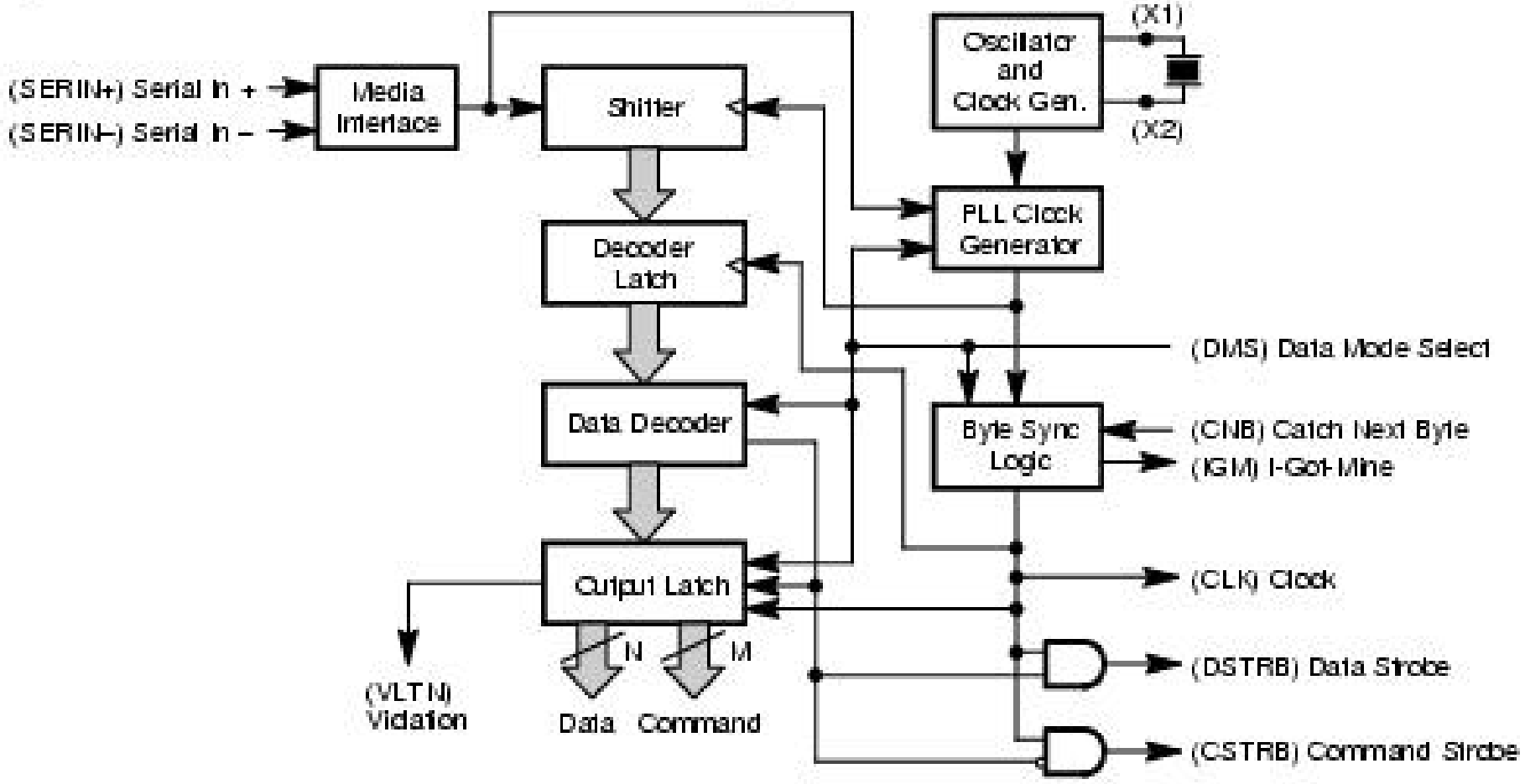
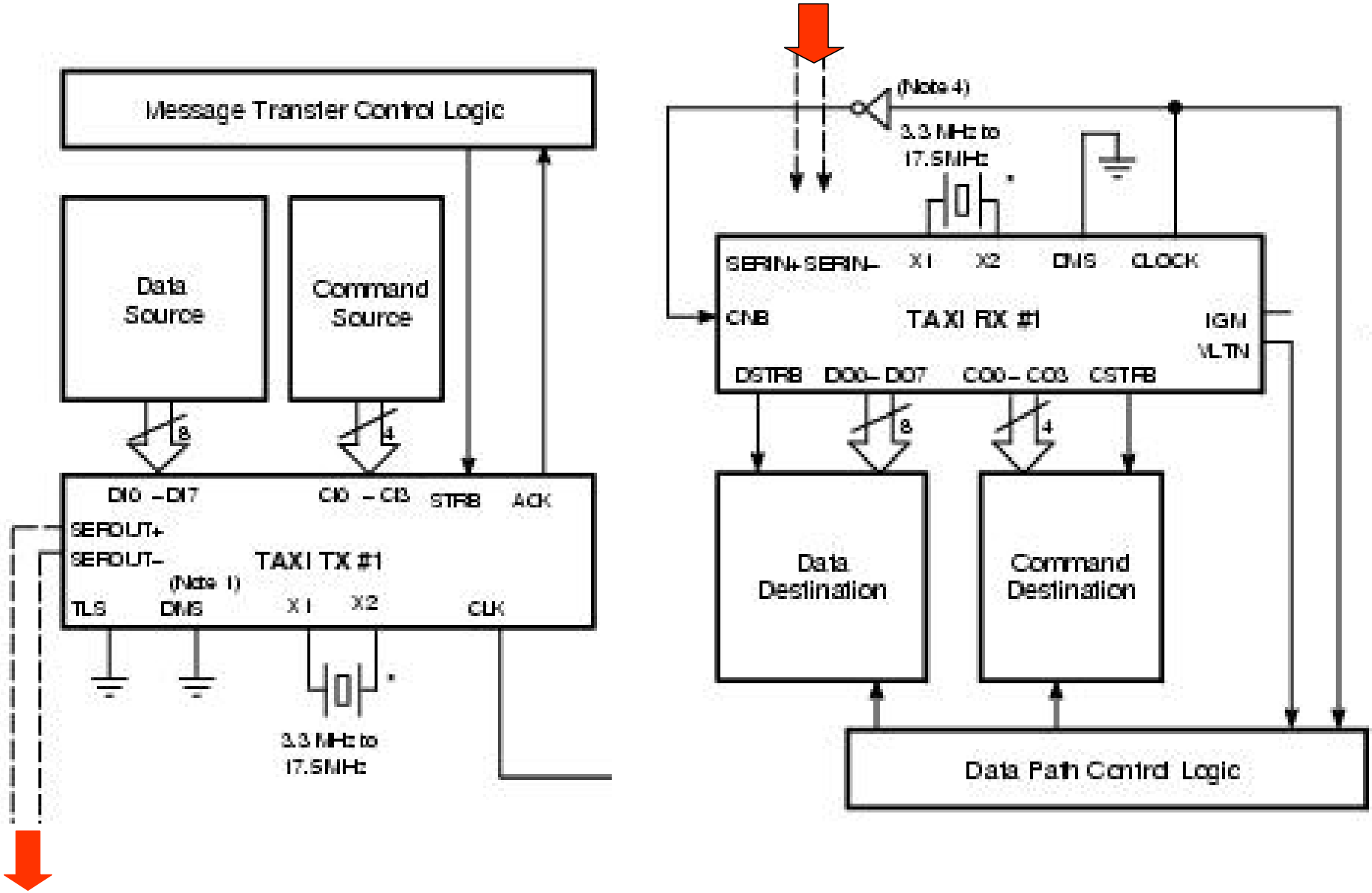


Figure 1-2 Am7969 TAXI Receiver Block Diagram



Note: N can be 8, 9, or 10 bits. Total of N + M = 12.

12330E-2


Message Transfer Control Logic
Data Source
Command Source
8
4
DI0 – DI7
CI0 – CI3
STRB
ACK
SEROUT+
SEROUT–
TAXI TX #1
(Note 1)
TLS
DMS
X1
X2
CLK
3.3 MHz to 17.5MHz
(Note 4)
3.3 MHz to 17.5MHz
SERIN+
SERIN–
X1
X2
DMS
CLOCK
CMB
TAXI RX #1
IGN
VLTN
DSTRB
DI00 – DI07
CI00 – CI03
CSTRB
8
4
Data Destination
Command Destination
Data Path Control Logic

Fiber sink (Taxi) block

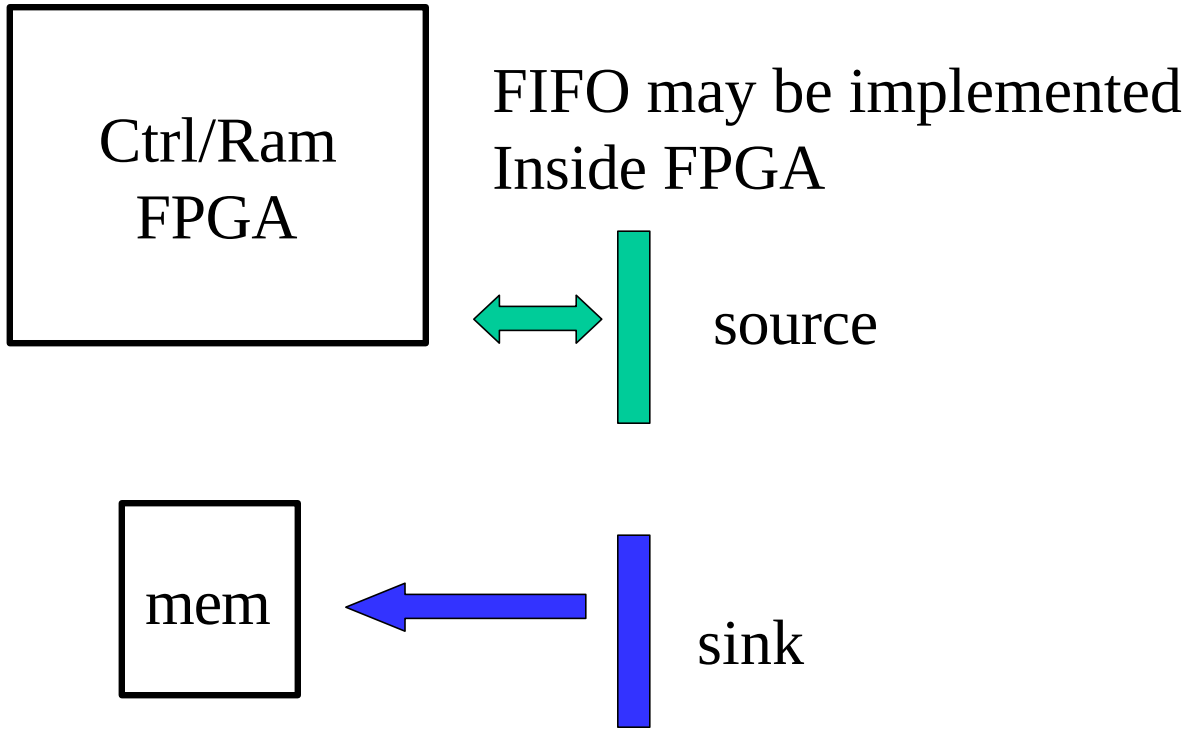
This one is hard, as the component (AM7969-175JC)
no longer exists.
Will have to find some (we need one for the sink
per board). Maybe from TRACER board spares? $25 each.

Bob DeMaat find some spares. Oct 4/01.

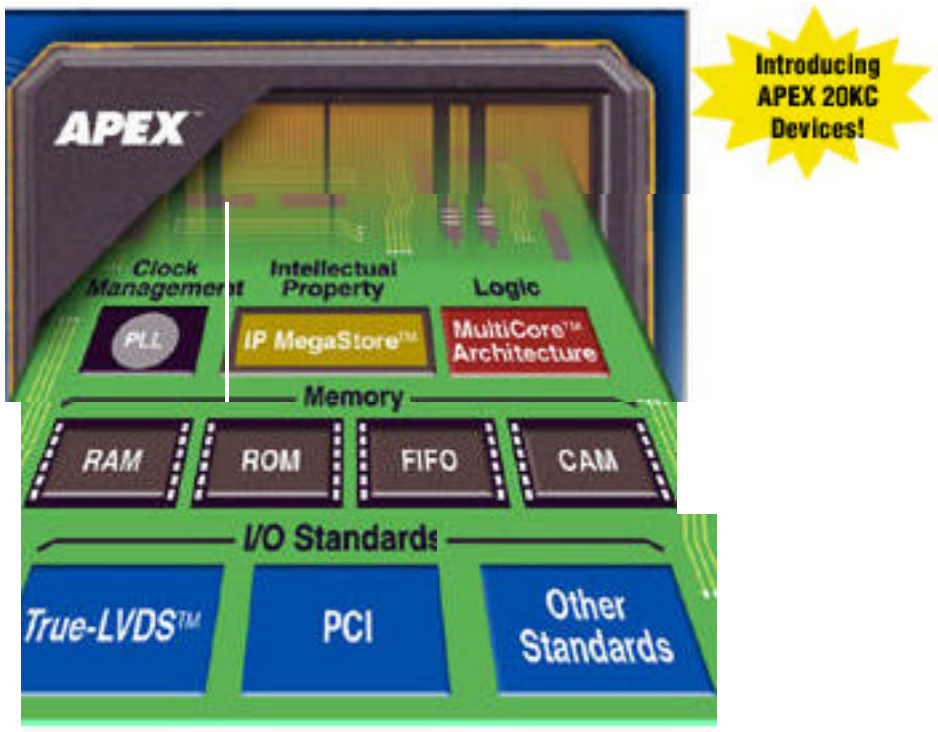
Also need optical transimitter: HFBR-1414T
Rx: HFBR-2415T
they are used on IsoList and Iso-Clique
check Reces as well.

SVT/XTRP, L1 source and sink block

Will be based on SVT Ghostbuster board design. Need to Find the details.

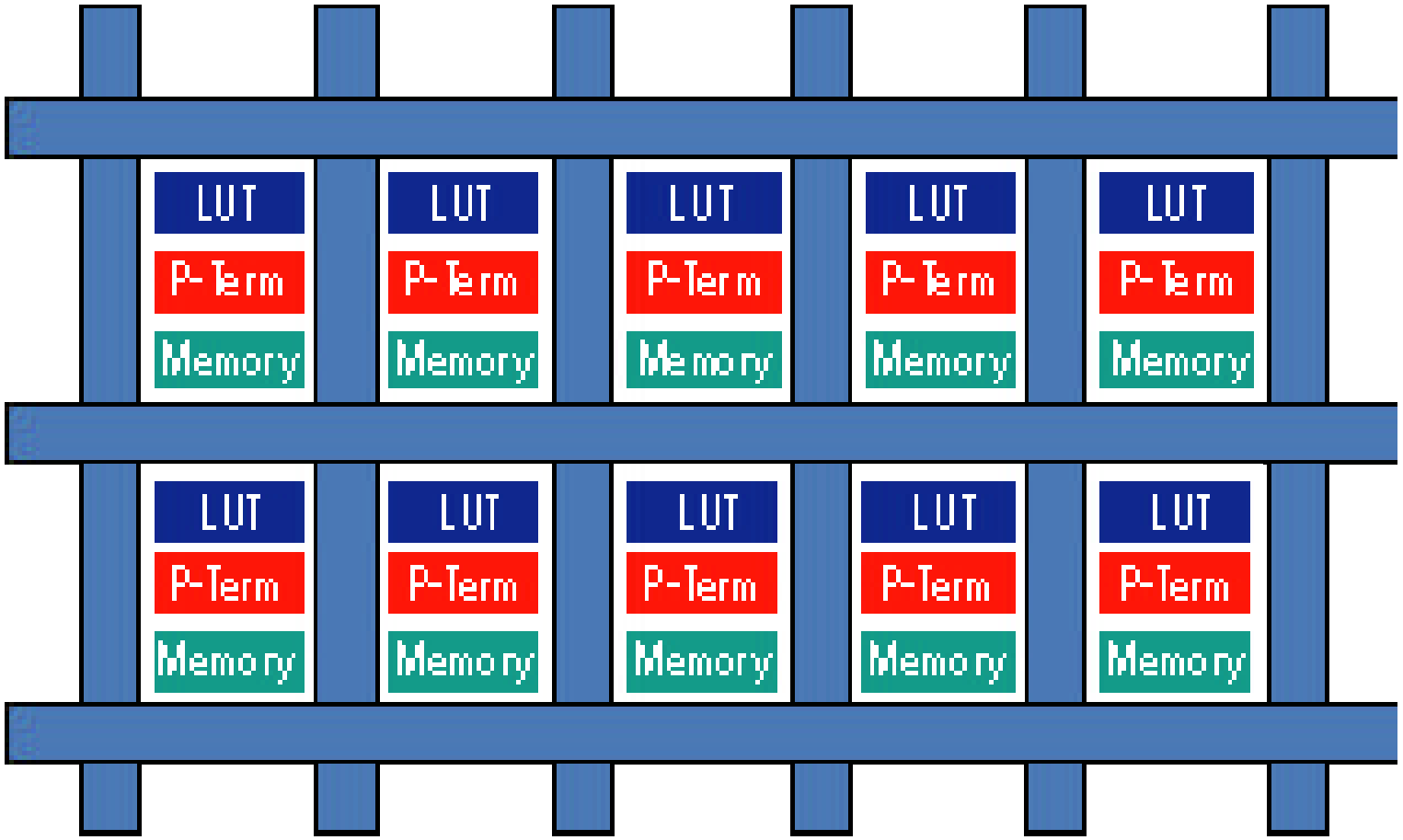


APEX 20K devices: System-on-a-Programmable-Chip solutions



Altera Local office: 847-240-0313?

Sales: Arrow. 630-285-6090(local). Mike Cannizzo


LUT
P-Term
Memory
LUT
P-Term
Memory
LUT
P-Term
Memory
LUT
P-Term
Memory
LUT
P-Term
Memory
LUT
P-Term
Memory
LUT
P-Term
Memory
LUT
P-Term
Memory
LUT
P-Term
Memory
LUT
P-Term
Memory

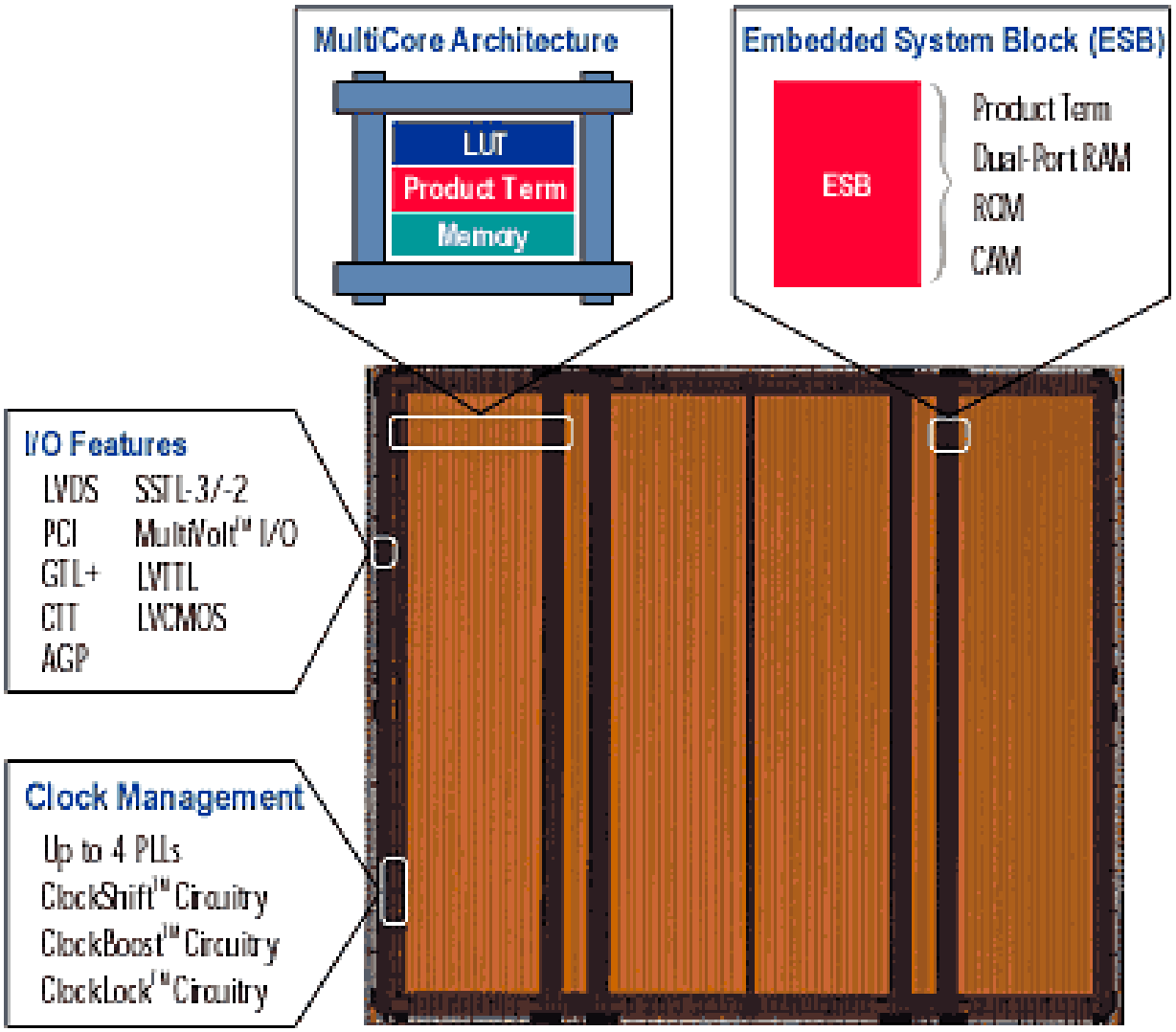

MultiCore Architecture
LUT
Product Term
Memory
Embedded System Block (ESB)
ESB
Product Term
Dual-Port RAM
ROM
CAM
I/O Features
LVDS
SSTL-3/-2
PCI
MultiVolt™ I/O
GTL+
LVTTL
CTT
LVCMOS
AGP
Clock Management
Up to 4 PLLs
ClockShift™ Circuitry
ClockBoost™ Circuitry
ClockLock™ Circuitry

Table 1. APEX 20K Device Features *Note (1)*

Feature	EP20K30E	EP20K60E	EP20K100	EP20K100E	EP20K160E	EP20K200	EP20K200E
Maximum system gates	113,000	162,000	263,000	263,000	404,000	526,000	526,000
Typical gates	30,000	60,000	100,000	100,000	160,000	200,000	200,000
LEs	1,200	2,560	4,160	4,160	6,400	8,320	8,320
ESBs	12	16	26	26	40	52	52
Maximum RAM bits	24,576	32,768	53,248	53,248	81,920	106,496	106,496
Maximum macrocells	192	256	416	416	640	832	832
Maximum user I/O pins	128	196	252	246	316	382	376

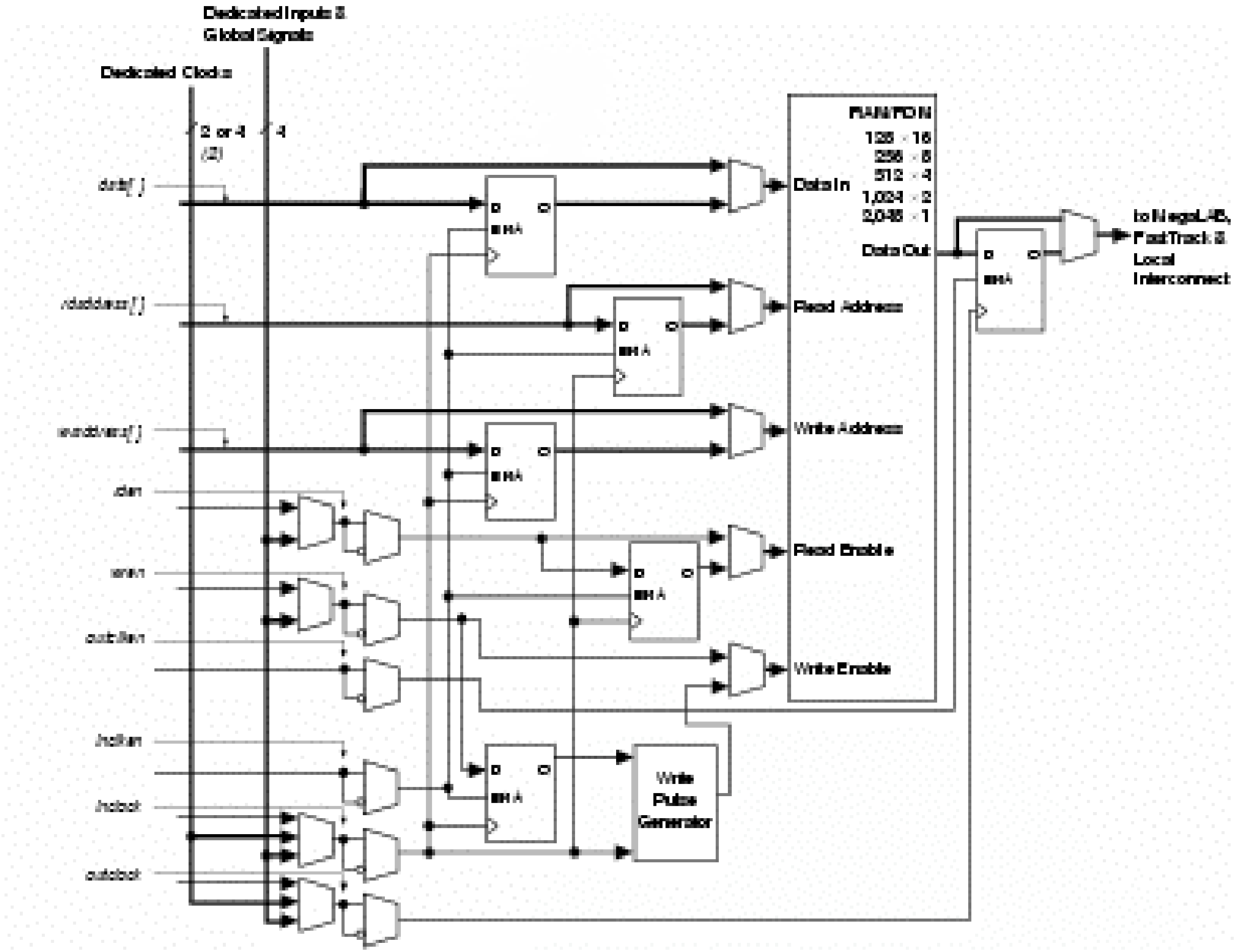
Table 2. APEX 20K Device Features *Note (1)*

Feature	EP20K300E	EP20K400	EP20K400E	EP20K600E	EP20K1000E	EP20K1500E
Maximum system gates	728,000	1,052,000	1,052,000	1,537,000	1,772,000	2,392,000
Typical gates	300,000	400,000	400,000	600,000	1,000,000	1,500,000
LEs	11,520	16,640	16,640	24,320	38,400	51,840
ESBs	72	104	104	152	160	216
Maximum RAM bits	147,456	212,992	212,992	311,296	327,680	442,368
Maximum macrocells	1,152	1,664	1,664	2,432	2,560	3,456
Maximum user I/O pins	408	502	488	588	708	808

Note to tables:

Table 4. APEX 20K QFP, BGA & PGA Package Options & I/O Count Notes (1), (2)						
Device	144-Pin TQFP	208-Pin PQFP RQFP	240-Pin PQFP RQFP	356-Pin BGA	652-Pin BGA	655-Pin PGA
EP20K30E	92	125				
EP20K60E	92	148	151	196		
EP20K100	101	159	189	252		
EP20K100E	92	151	183	246		
EP20K160E	88	143	175	271		
EP20K200		144	174	277		
EP20K200E		136	168	271	376	
EP20K300E			152		408	
EP20K400					502	502
EP20K400E					488	
EP20K600E					488	
EP20K1000E					488	
EP20K1500E					488	

Figure 20. ESB in Read/Write Clock Mode *Note (1)*



Ctrl/RAM FPGA for fiber links

Current thinking is to use two APEX 20K FPGA (mainly for flexibility). One for fiber links and the other for SVT/XTRP and L1.

EP20K200(E) (~$190 each) Main features we need:

--- embedded system block (ESB) used to implement memory functions: FIFO, RAM etc; 52 ESBs total.

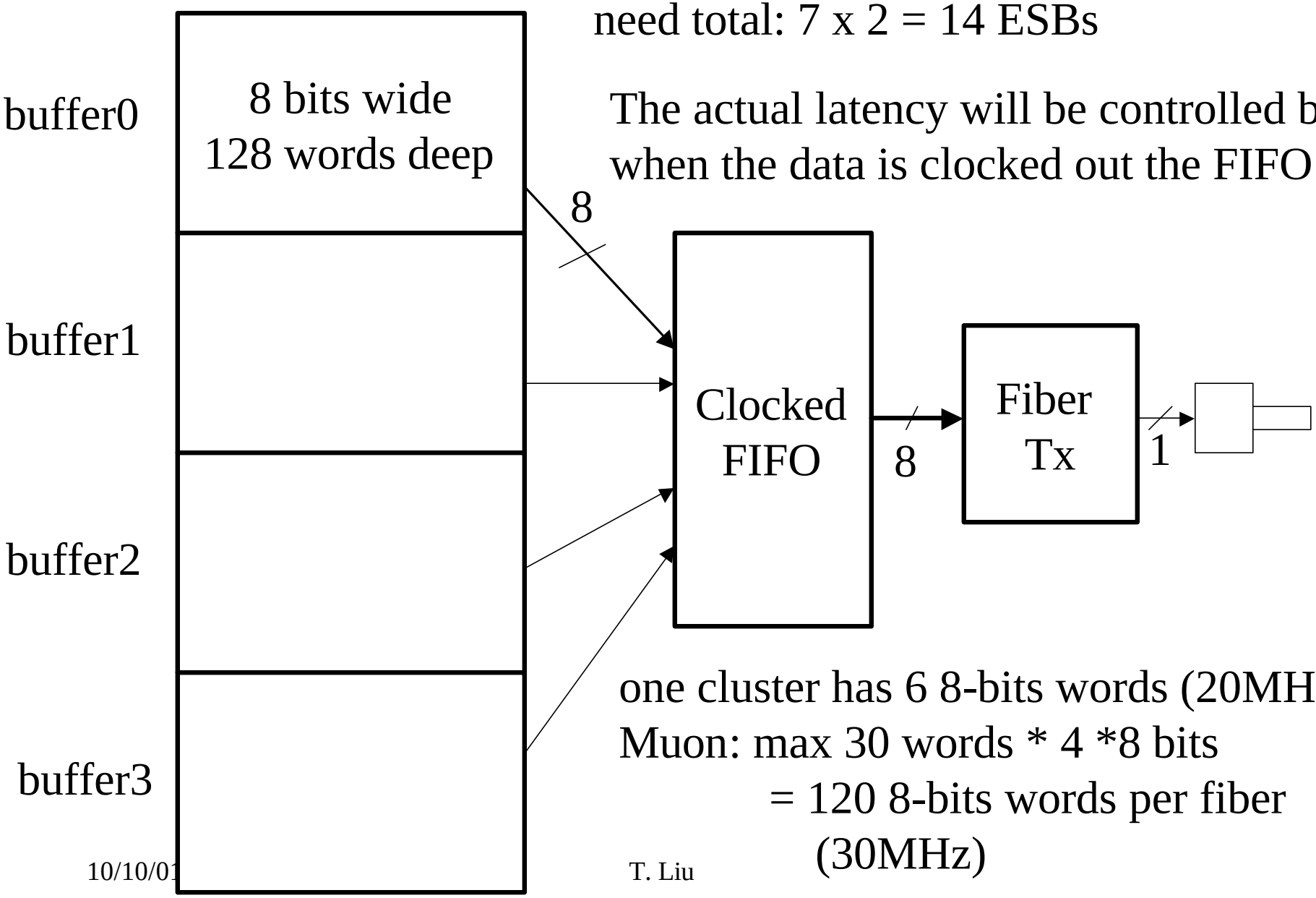
--- 106K bits maximum total RAM bits available;

--- each ESB can be configured in 256 x 8 or 128 x 16 memory size; two ESBs can be used to form 512 x 8 bits RAM block. can use 512/4 = 128 bytes per buffer number. Enough to store typical event per fiber channel. Need 7 total: 7 x 512x8 = 28.7K bits; use 7x2 = 14 ESBs.

--- flexible clock management with up to four PLLs (E); dedicated clock and input pins: 6/8(E)

Two ESBs can form 512 x 8 bits RAM, can be used for one fiber out

need total: 7 x 2 = 14 ESBs



The actual latency will be controlled by when the data is clocked out the FIFO

one cluster has 6 8-bits words (20MHz)

Muon: max 30 words * 4 *8 bits

= 120 8-bits words per fiber (30MHz)

Example: CLIST cluster information from one LOCOS
6 8-bits words per cluster on one fiber input, arriving 50ns apart

train no.	I	II	III	IV	V	VI
sig_0	1	em(5)	1	had(5)	1	crate_sel
sig_1	L1AB(0)	em(6)	passbit(0)	had(6)	phi(0)	ntow(0)
sig_2	L1AB(1)	em(7)	passbit(1)	had(7)	phi(1)	ntow(1)
sig_3	em(0)	em(8)	had(0)	had(8)	eta(0)	ntow(2)
sig_4	em(1)	em(9)	had(1)	had(9)	eta(1)	ntow(3)
sig_5	em(2)	em(10)	had(2)	had(10)	eta(2)	ntow(4)
sig_6	em(3)	em(11)	had(3)	had(11)	eta(3)	ntow(5)
sig_7	em(4)	em(12)	had(4)	had(12)	eta(4)	ntow(6)

Data format from Monica.

1 CLIQUE connection (via 10 pin twisted ribbon cable)

The LVDS signals are driven by a 16 nsec clock which is a divided-by-8 copy of the 132 nsec CDF clock:

pin 1 BUF_DONE(0)+
pin 2 BUF_DONE(0)-
pin 3 BUF_DONE(1)+
pin 4 BUF_DONE(1)-
pin 5 CRSUM_SEND+ (not received by CLIST)
pin 6 CRSUM_SEND- (not received by CLIST)
pin 7 EVENT_DONE*+
pin 8 EVENT_DONE*-
pin 9 unused
pin 10 unused

The time of EVENT_DONE* with respect to the last cluster found in the event is fixed.

LVDS output

8-bits wide cluster data:
Em(4:0), buff(1:0),1
Em(12 : 5)
Had(4:0), pass(1:0),1
Had(12 : 5)
Eta(4:0), phi(1:0), 1
Ntow(6:0), crate_sel

(assume this is the last cluster for the event ➔)

Evt_done, buff(1:0)

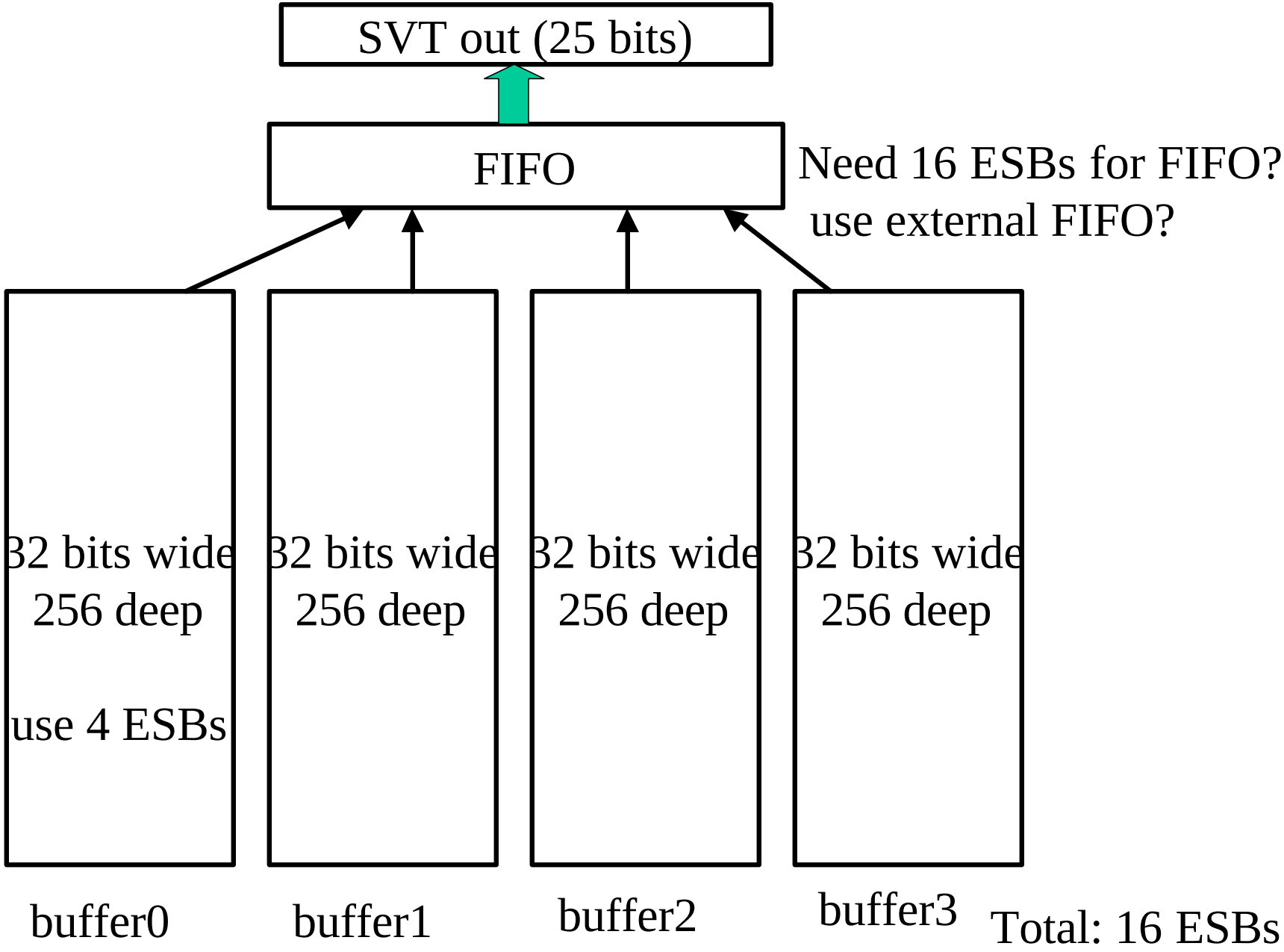
CLIQUE control word

Ctrl/RAM FPGA for SVT/XTRP and L1

EP20K200(E) Main features:

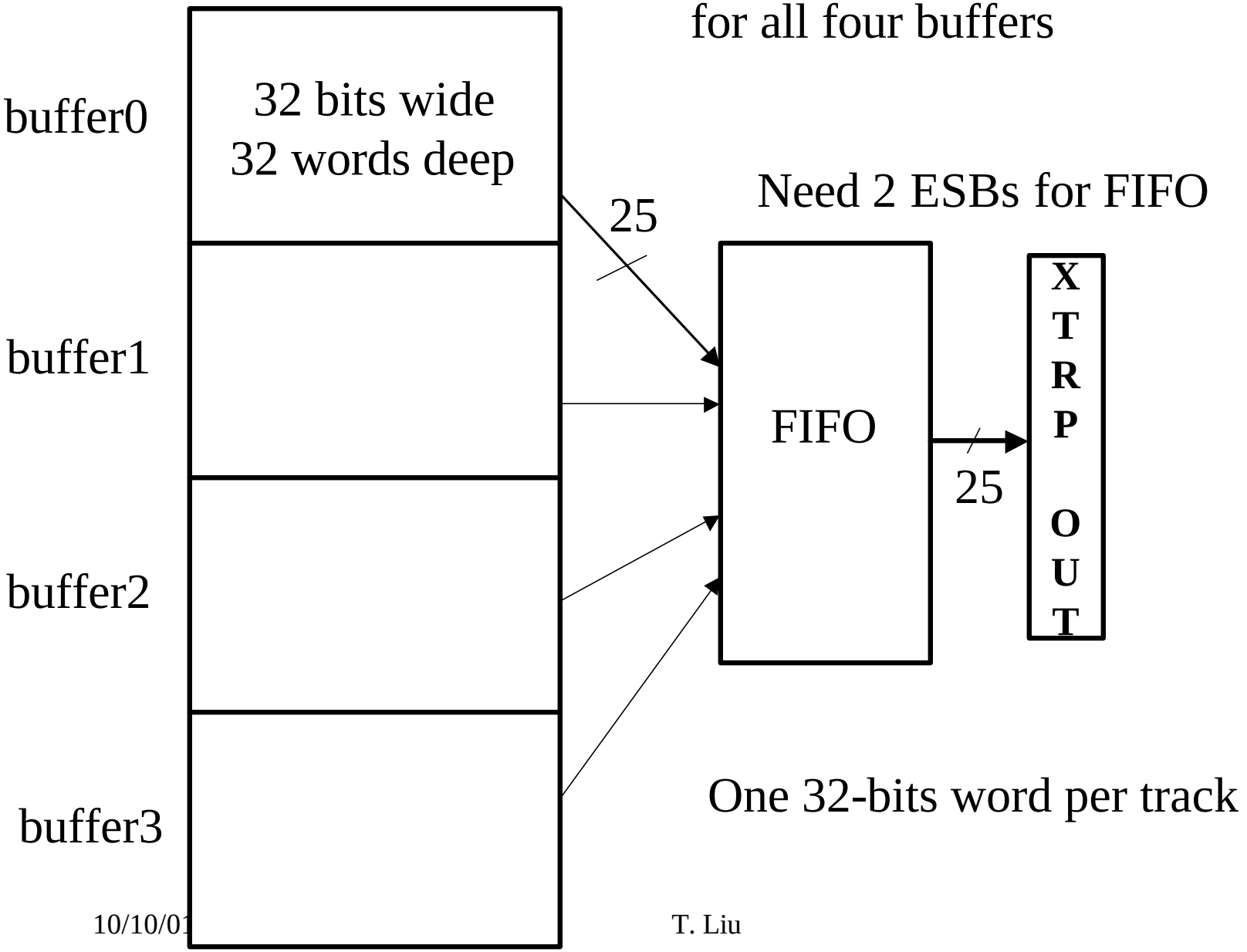
--- each ESB can be formed in 256 x 8 or 128 x 16 RAM size;

* use 4 ESBs to form 256 x 32 bits RAM block for
  SVT data (25bits wide, 7 words per track) for
  each buffer. Can store 256/7 = 36 tracks data per buffer.
  Need 4 x 4 = 16 ESBs for all four buffers;
* use 2 ESBs to form 128 x 32 bits RAM for XTRP data
  (25 bits wide, 1 word per track). Can store 128/4 = 32
  track words per buffer;
* use 2 ESBs to form 64 x 64 bits RAM for L1 data;
  one 64-bits form one event worth data.
  64/4 = 16 L1 words available,
  but only need one per buffer number.
* total ESB needed: 16 (SVT) + 2 (XTRP) + 2 (L1) = 20,
  total ESB available: 52.

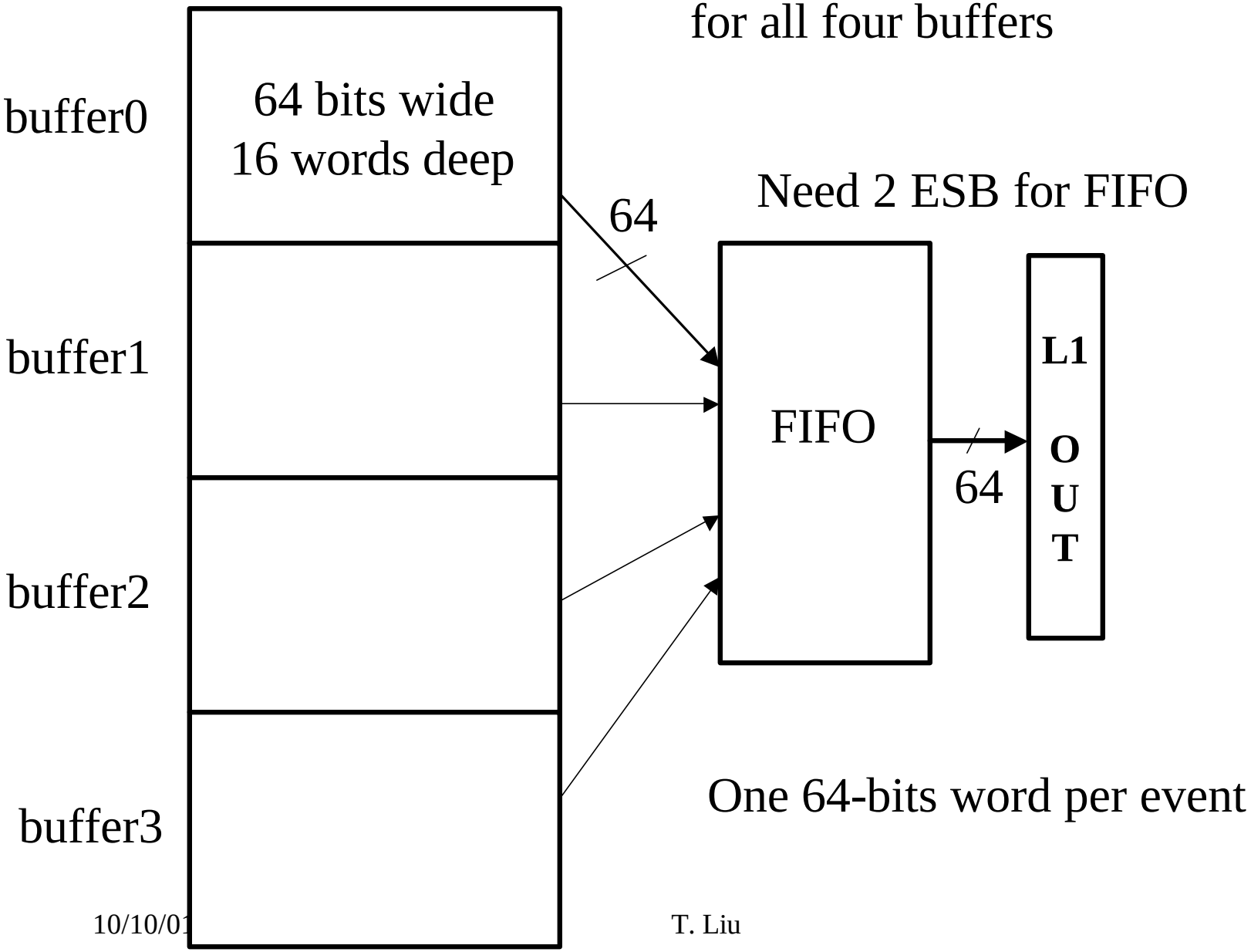


7 25-bits words per track, can store up to 36 tracks data

Two ESBs can form 128 x 32 bits RAM, can be used for XTRP out for all four buffers



Two ESBs can (?) form 64 x 64 bits RAM, can be used for L1 out for all four buffers



One 64-bits word per event

What if L1As coming very close (132ns apart)?

Well, the only data arrives to L2 early is L1 bits (within 1 or 2 cdfclk?). Everyone else will be much later.

The idea is to queue the L1As as they arrive in Ctrl/RAM FPGA. Upon receiving the first L1A, the 64 bits L1 data will be written into output FIFO within one cdfclk, and next L1A L1 data can be written into output FIFO right away. So this shouldn't be a problem. Other data path latency will be on the order of us, should have plenty of time to write the data into output FIFO. Should check every data path though (event size dependent). If there is a problem, increase the write clock speed from RAM to output FIFO.

Need to find out typical latency for each data path, and the range.

Data sink memory implementation

Desire deep memory for recording:

-- in self test mode, record each data source output data;
-- record real data source in situ;
-- recording can be stopped by any backplane signals,
do we need external trigger?
-- LA output pins;
-- readout via VME;
-- if use 16 bits address memory, running at 30 MHz,
one can store 32K x 30ns ~ 1000 us history.

separate memory chip can be used. Configured as
a cyclic latency buffer continuously written at the same rate
data is received. At any given time, 1 ms(?) history can be
recorded.

I/O pin counts for Ctrl/Ram FPGA1 and FPGA2

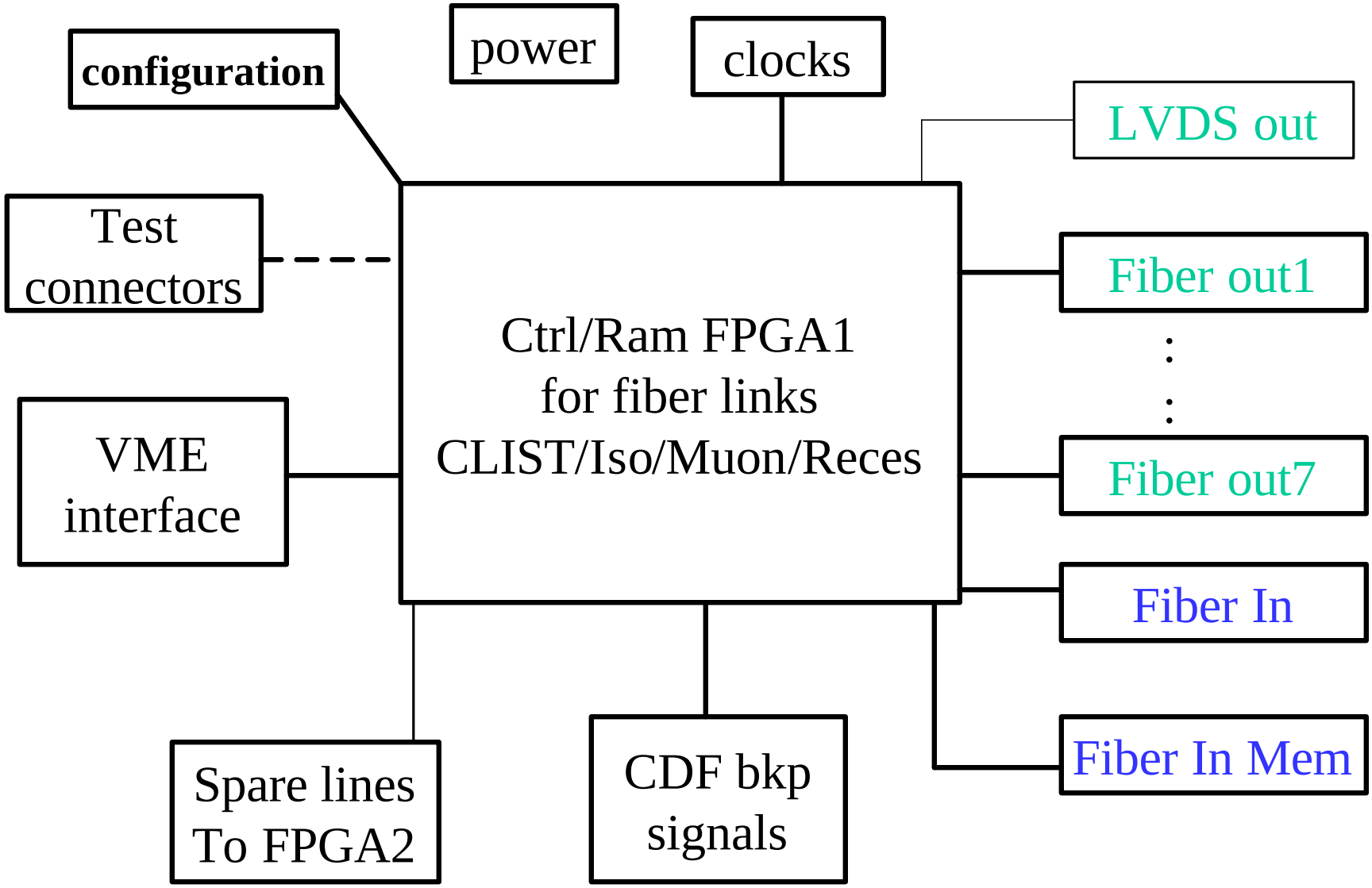
Note that EP20K200 maximum user I/O pins: 382

EP20K200E	376
EP20K400	502
EP20K400E	488

First round: worse case I/O pin counts.

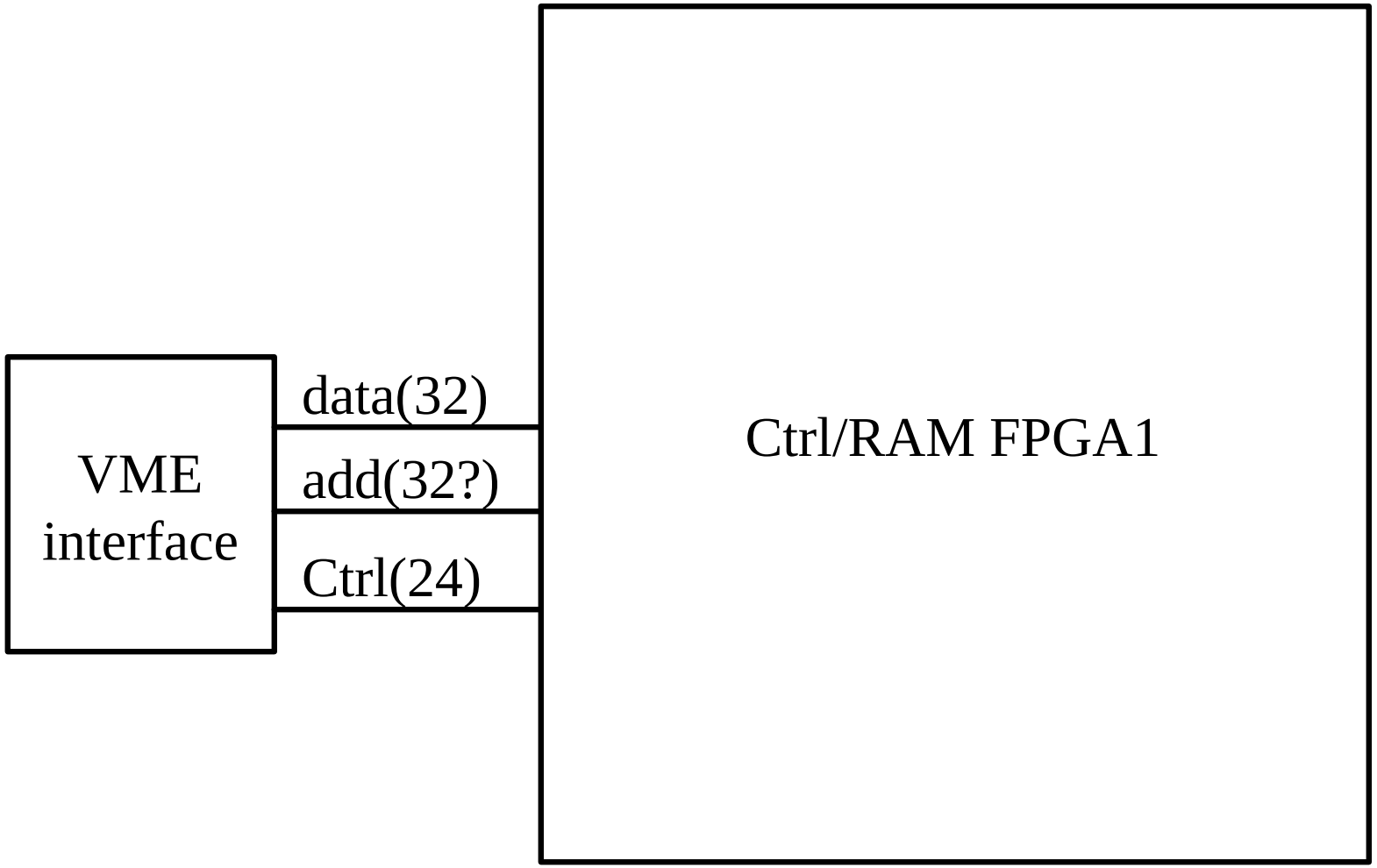
Need more careful I/O pin count later, and find out whether we will need to use BGA or not.

Ctrl/RAM FPGA1 for fiber output block diagram

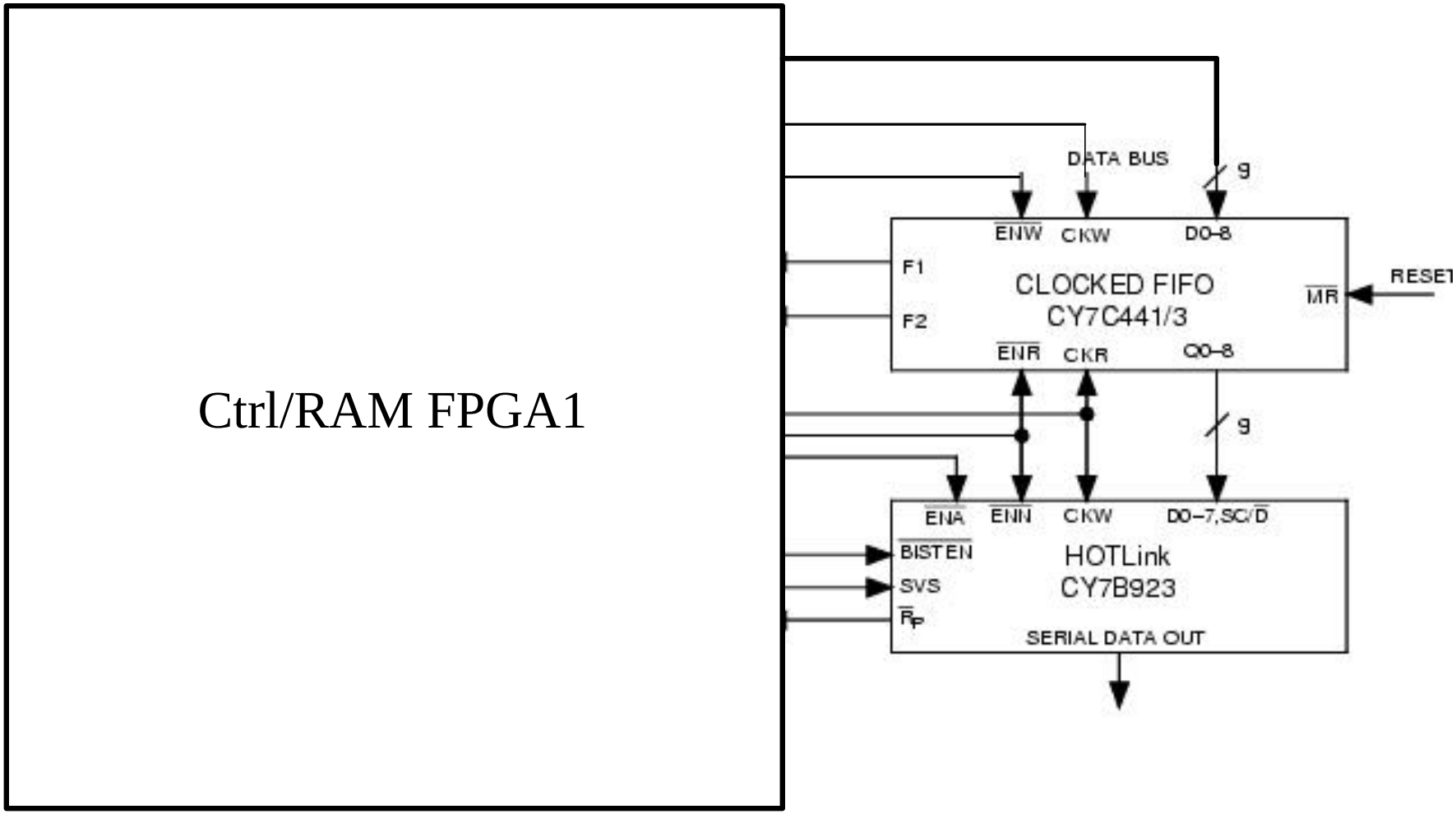


VME interface I/O pins: 32+32+24 = 88 maximum

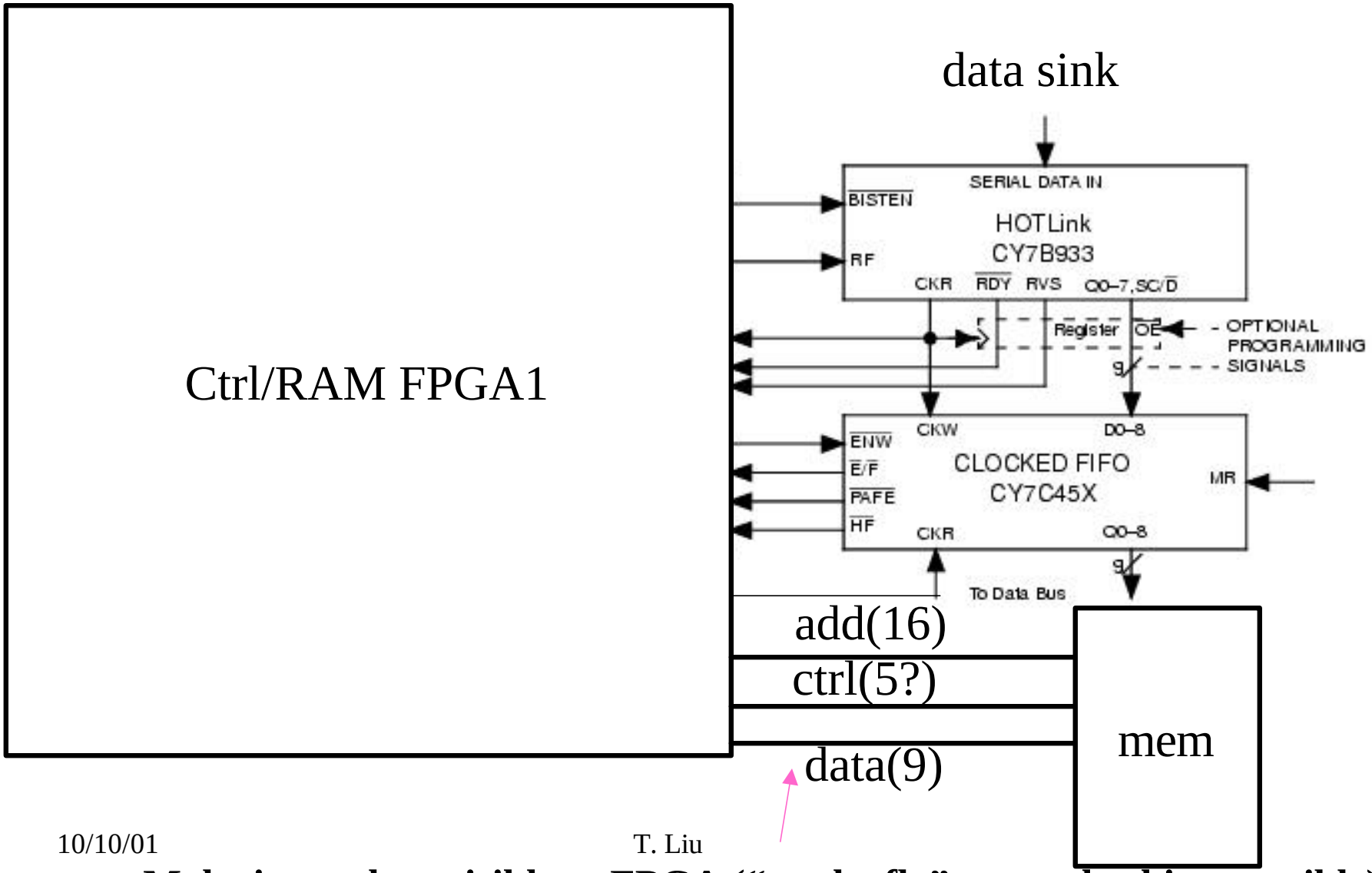
actual need will be less…(slave VME)



Fiber Out interface I/O pins: 20 maximum x 7 = 140

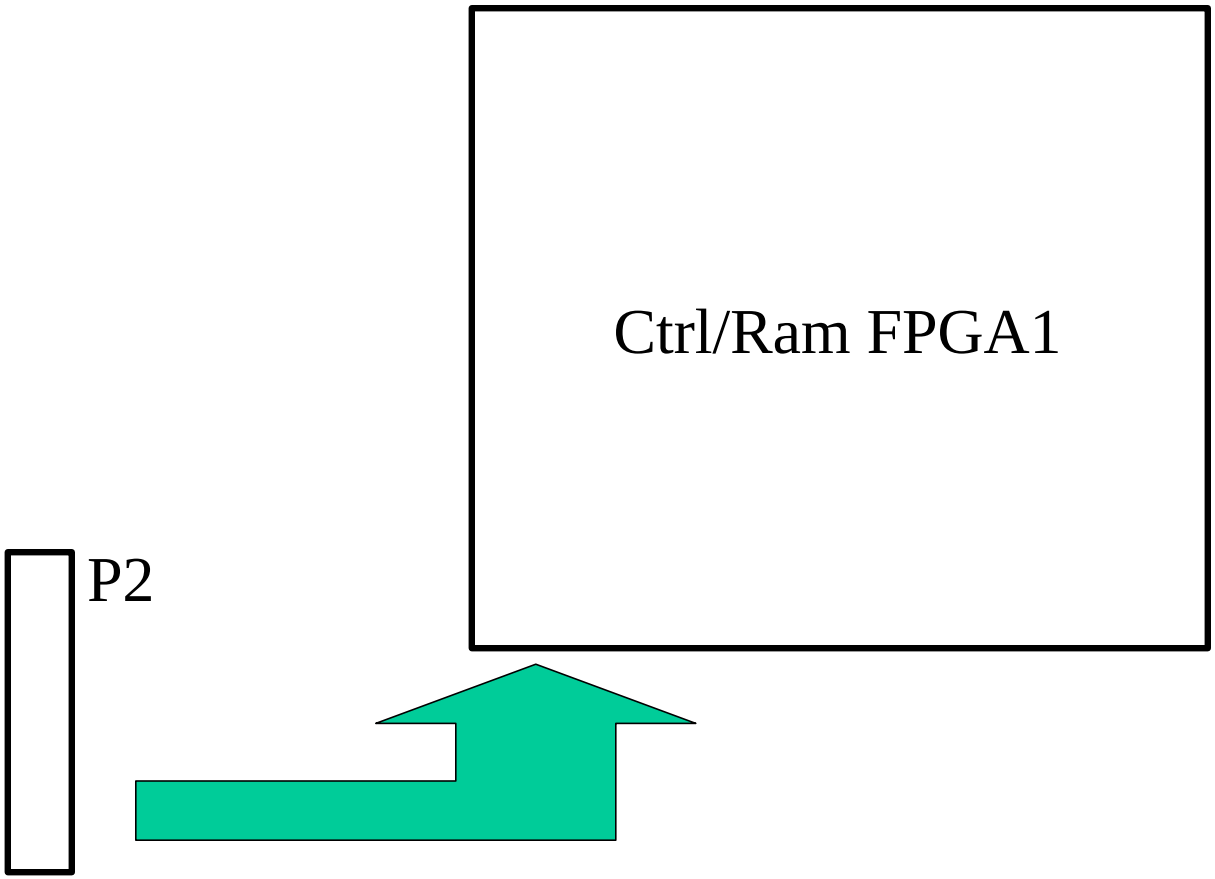


Fiber in (mem) interface I/O pins: 11+16+9+5?= 41maximum



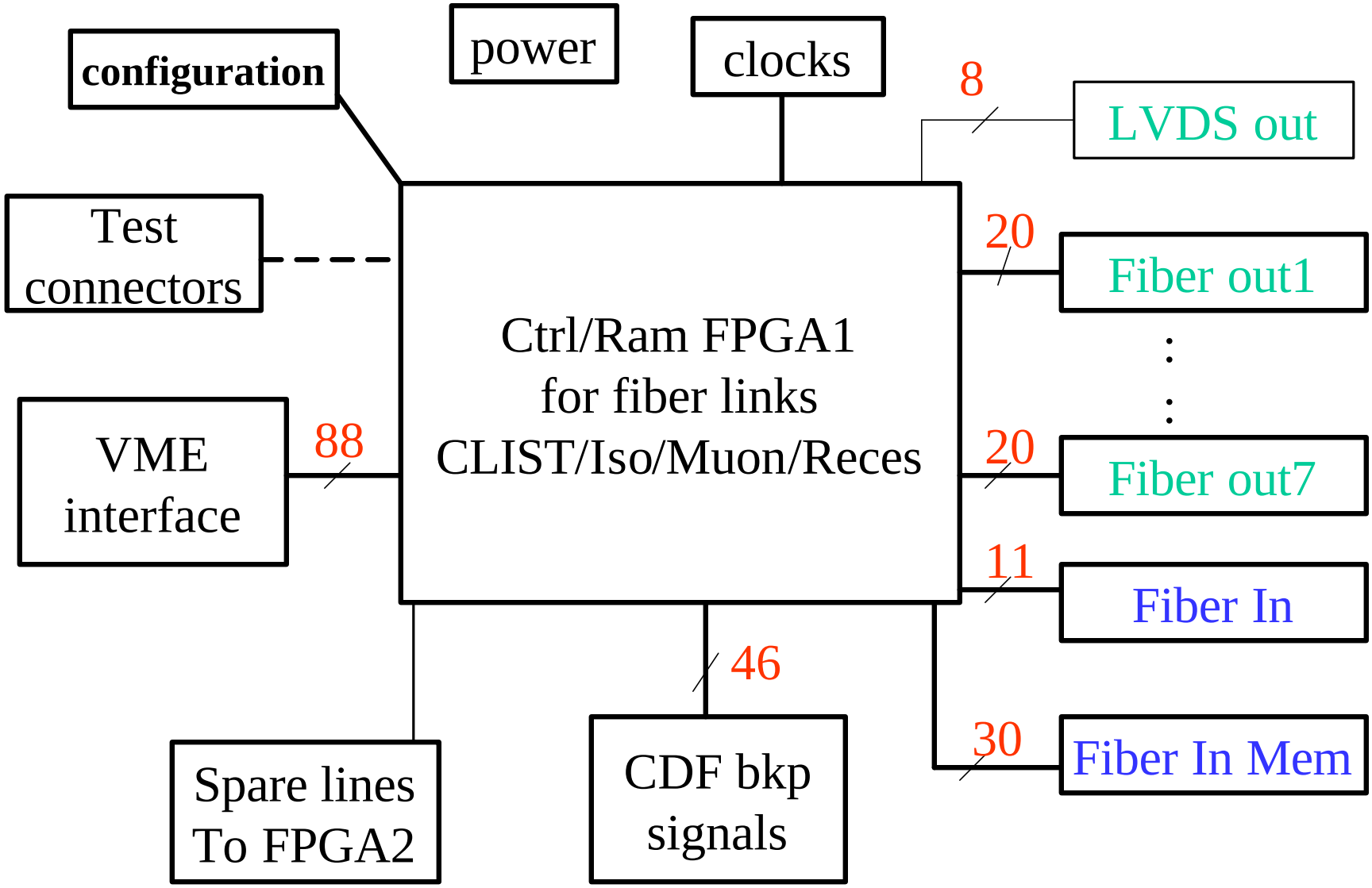
Make input data visible to FPGA (“on the fly” error checking possible)

Interface with CDF backplane signals: 46 maximum



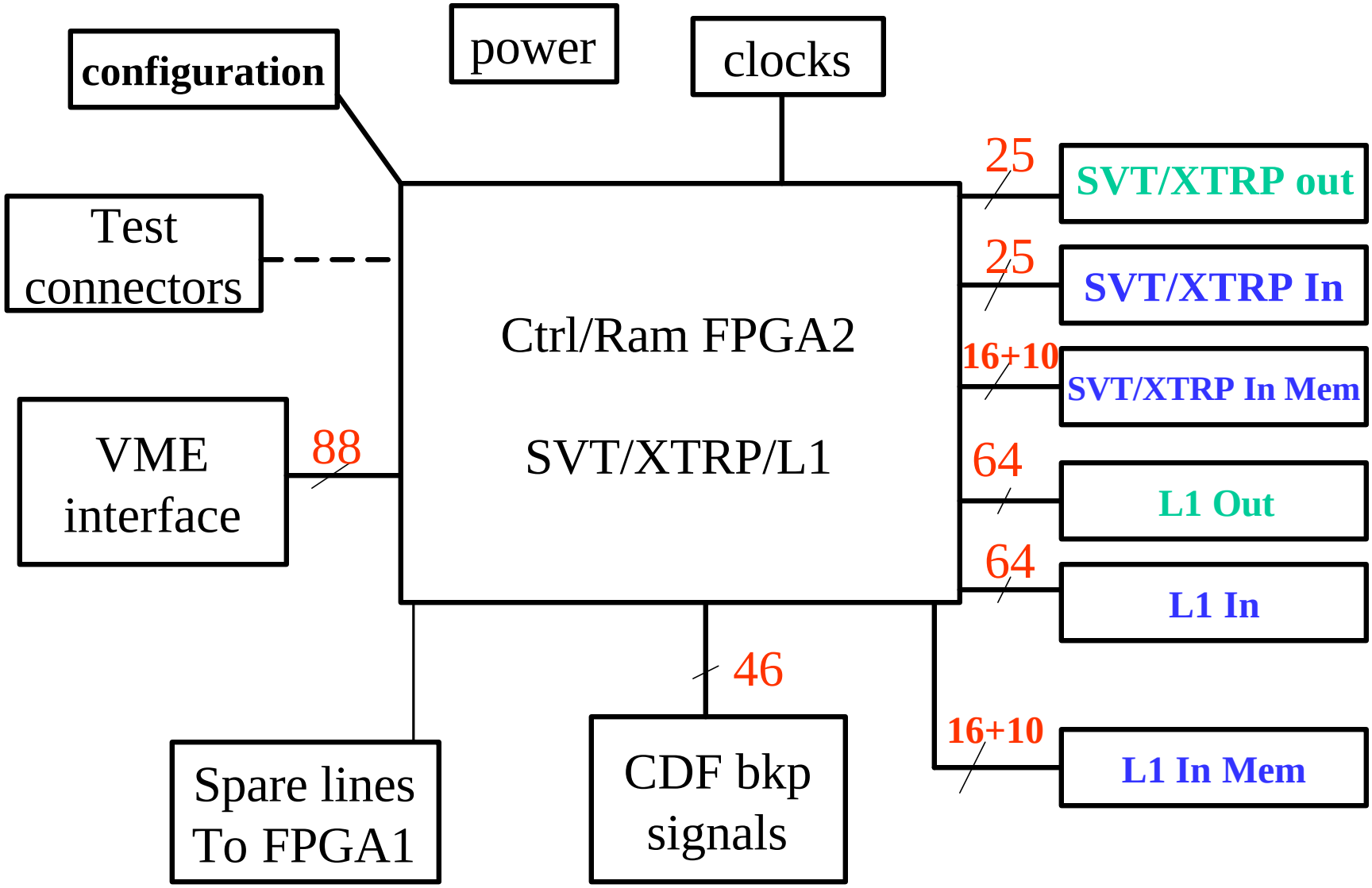
CDF signals from P2 backplane: 46 total, but many not needed

Ctrl/RAM FPGA1 I/O pins(worse case):



Maximum total needed: 323/382, add 32 for test pins.

Ctrl/RAM FPGA2 for SVT/XTRP/L1 I/O pins:



Total (worse case) I/O pins: 364/382 check details!

How many boards needed?

With this design:

CLIST input: need one test board with hotlink;

IsoList input: need one test board with Taxi;

SVT/XTRP/L1: two fully loaded boards;

so to drive SVT/XTRP/L1/CLIST/Iso, we only need two fully loaded test boards (one Taxi, one Hotlink);

Muon input: need 3 test boards (16 fiber hotlink inputs);

Reces input: need 2 to test one Reces (12 fiber Taxi inputs),
need 2x4=8 to drive all 4 Reces inputs.
those boards can be partially loaded (fibers only).

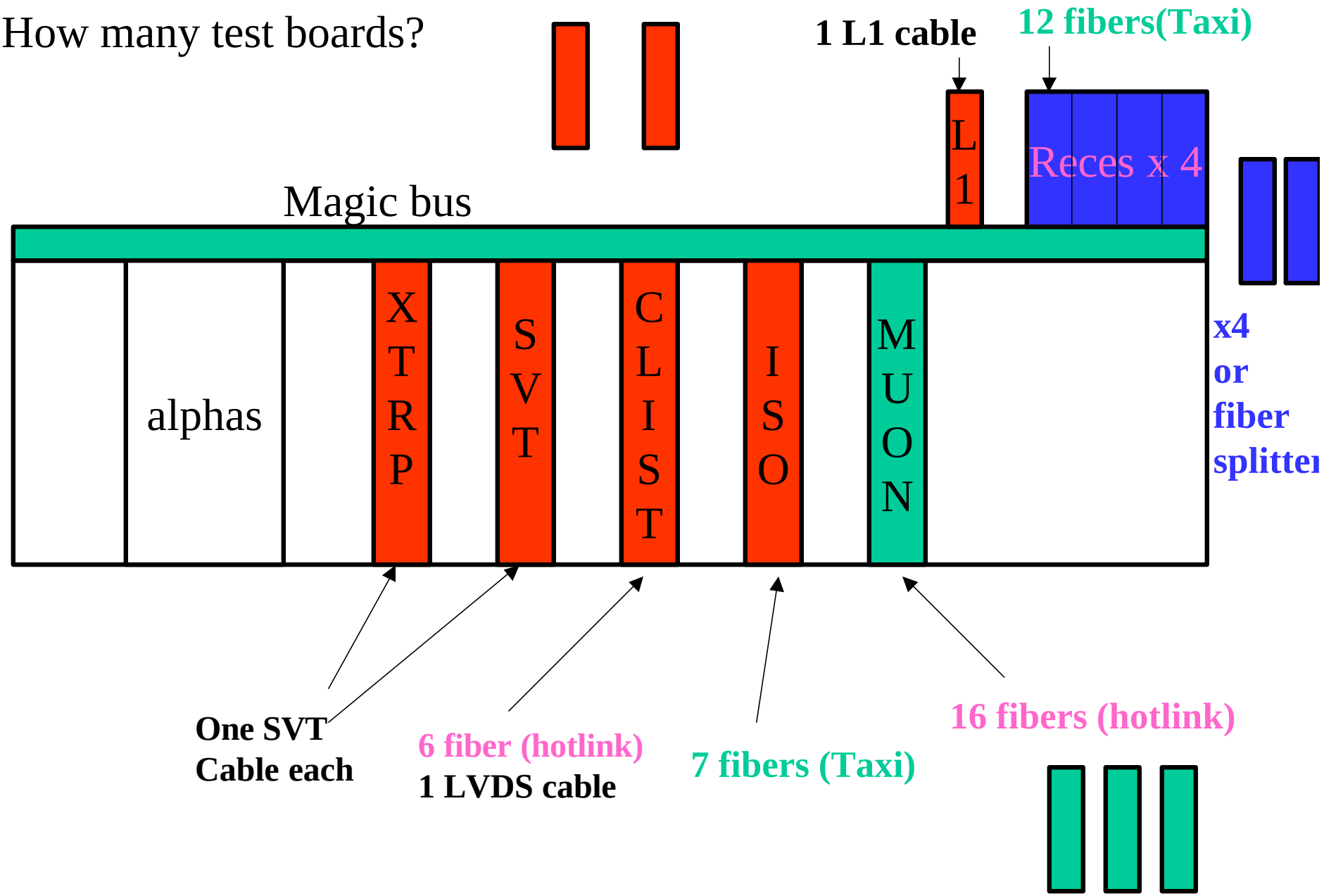
Plan: initially fully load 4 boards with hotlink, 2 with TAXI.
Can be used to drive either:
L1/SVT/XTRP/CLIST/Iso/Muon, or
L1/SVT/XTRP/CLIST/Iso/one Reces

Have to keep the cost per board low ($2-3k?): $3k x 6 =$18K.

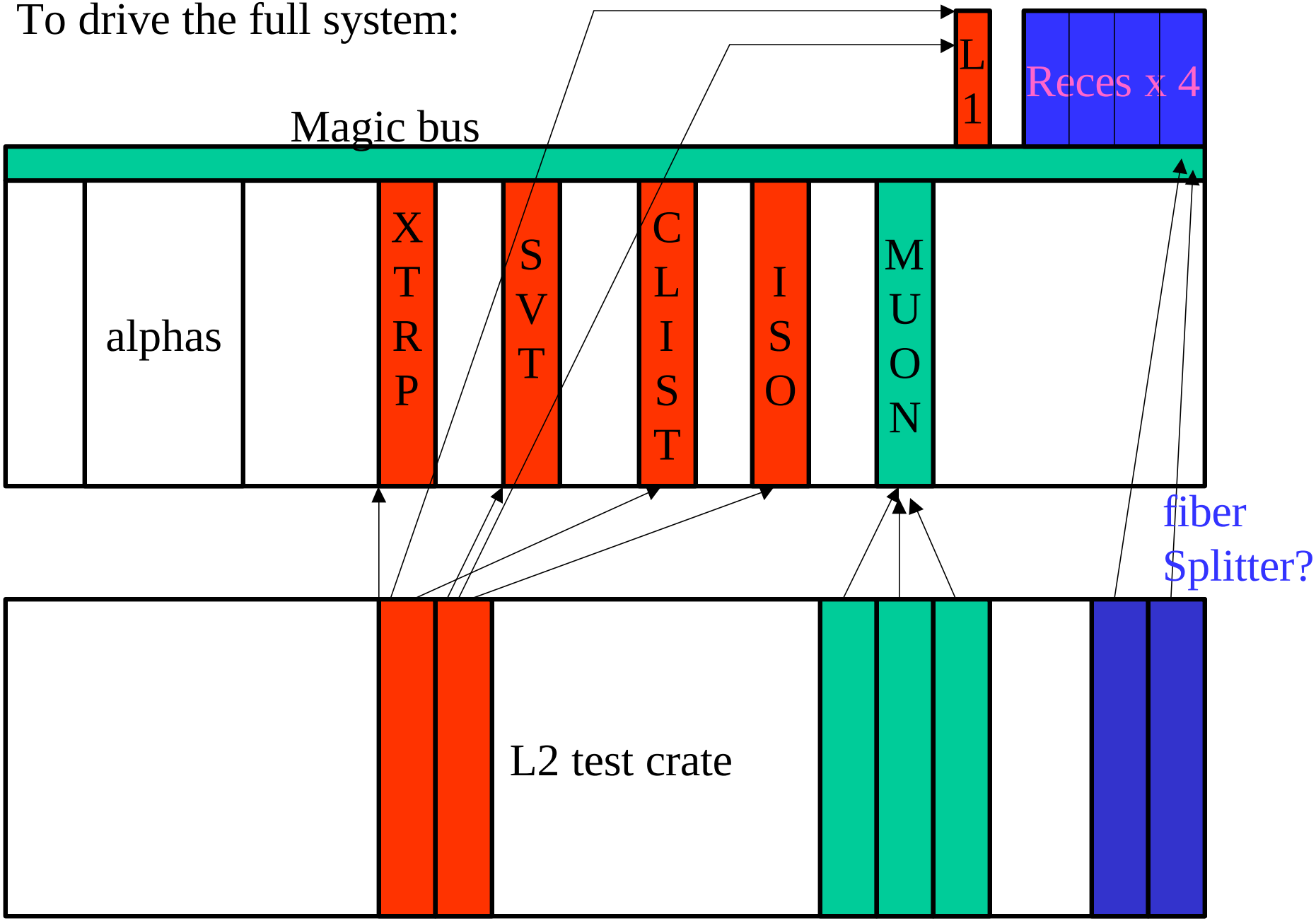
To be able to drive the full system: 2 + 3 (hotlink) + 8 (Taxi) =13.

(fiber splitter?) 4 hotlink, 9 Taxi

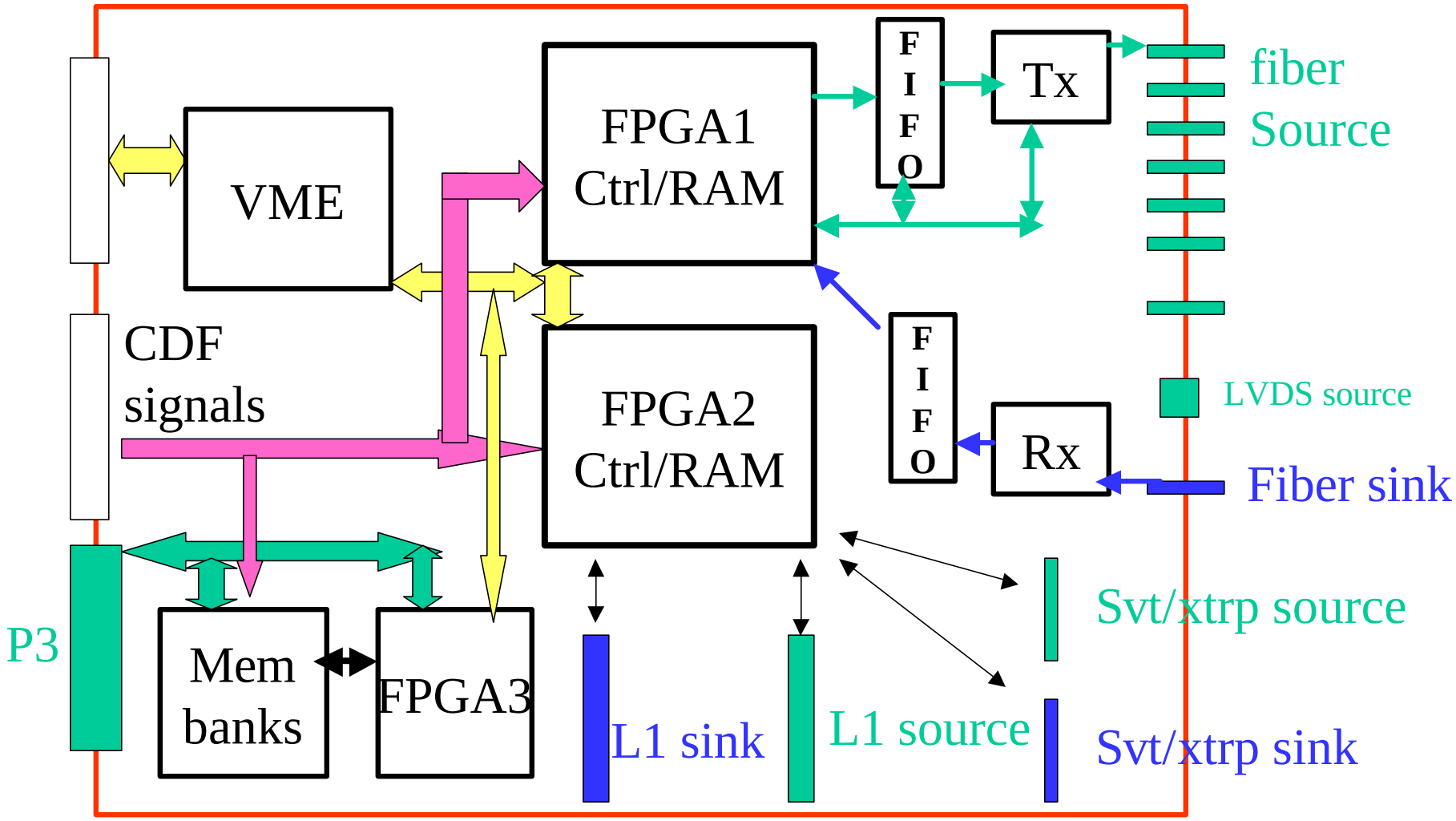
How many test boards?



To drive the full system:

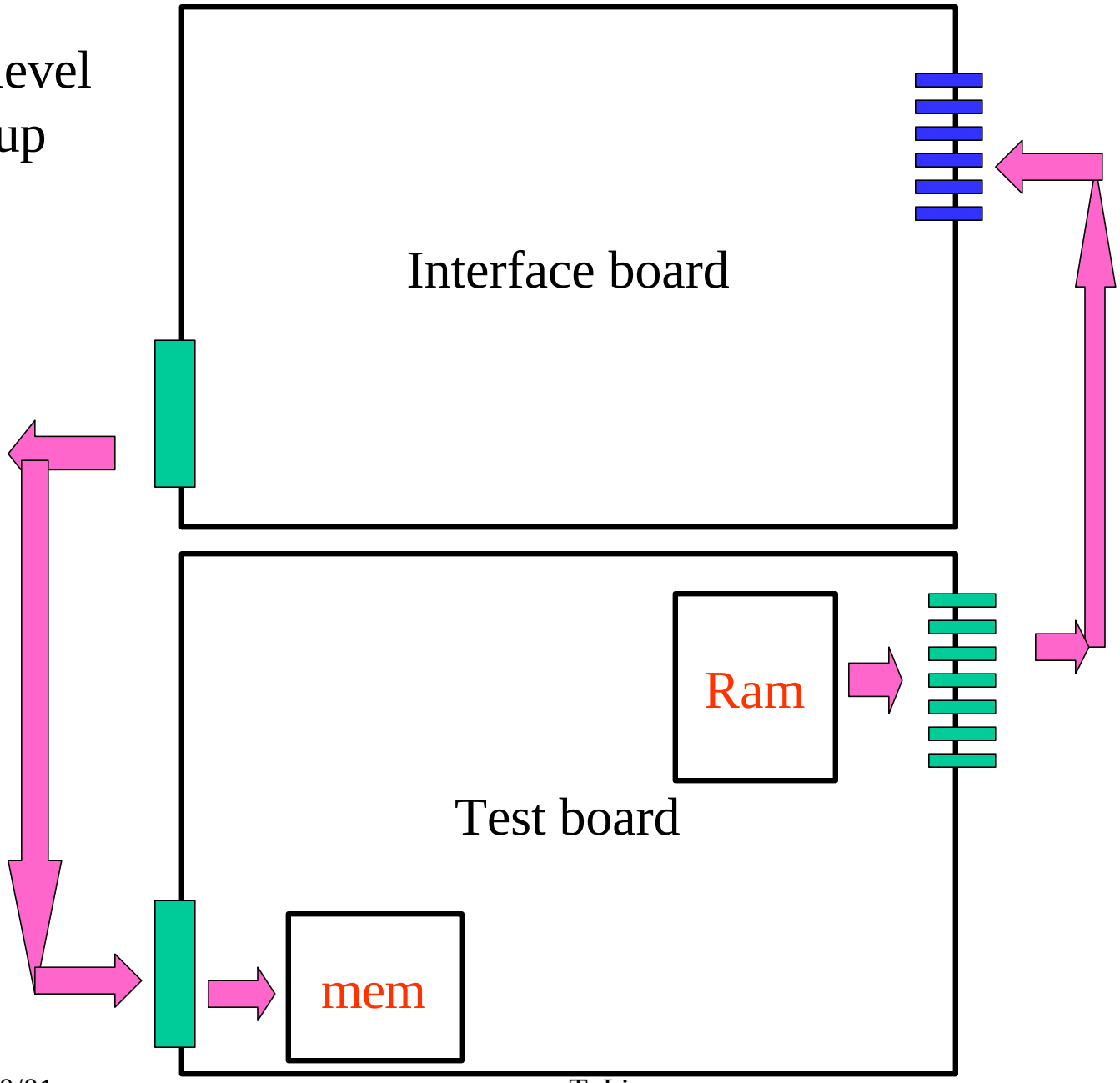


* optional: spy/control magic bus and P2 signals?



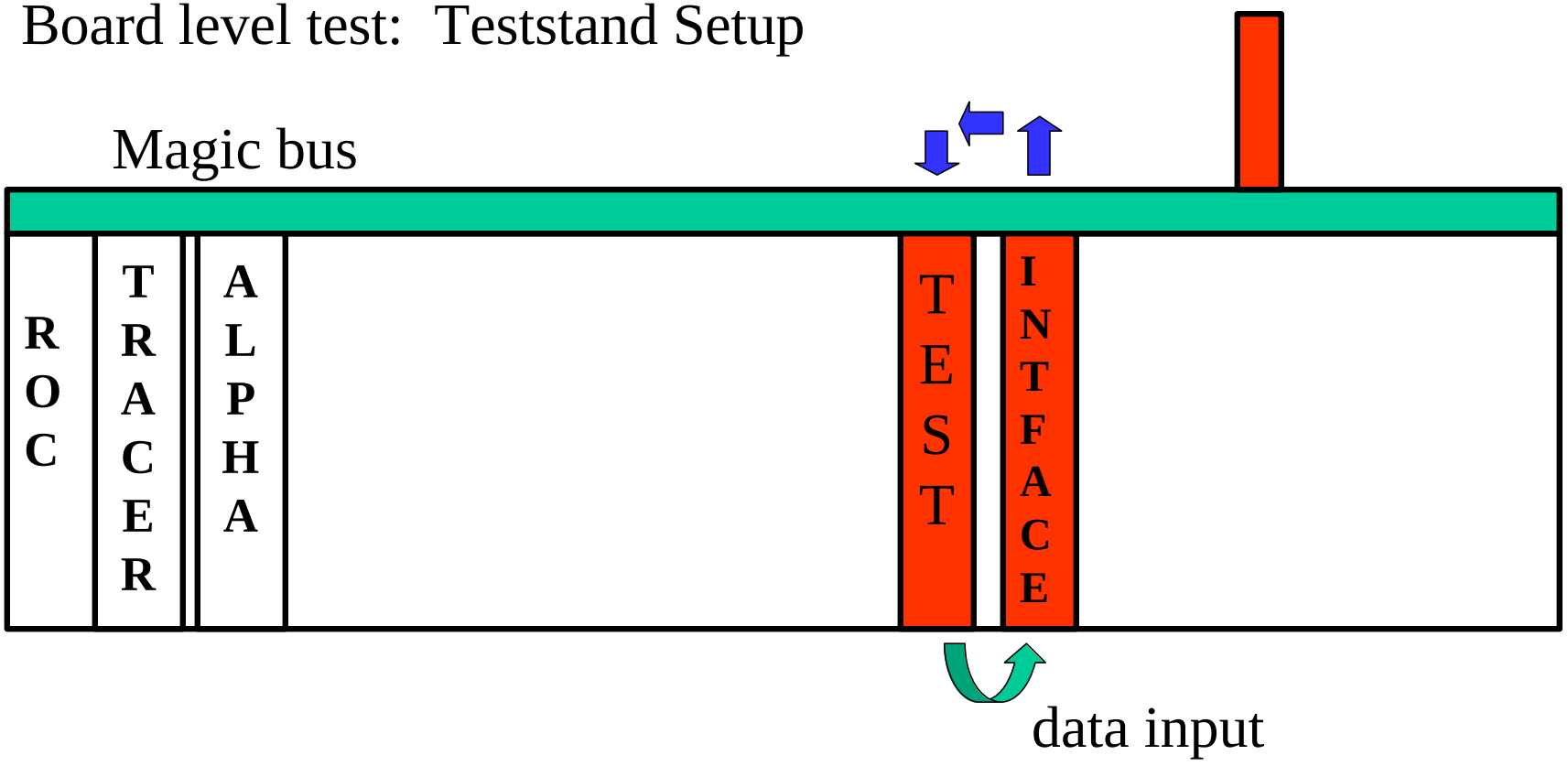
Magic bus interface: as target and master, or simply spying Magicbus data

Board level test setup



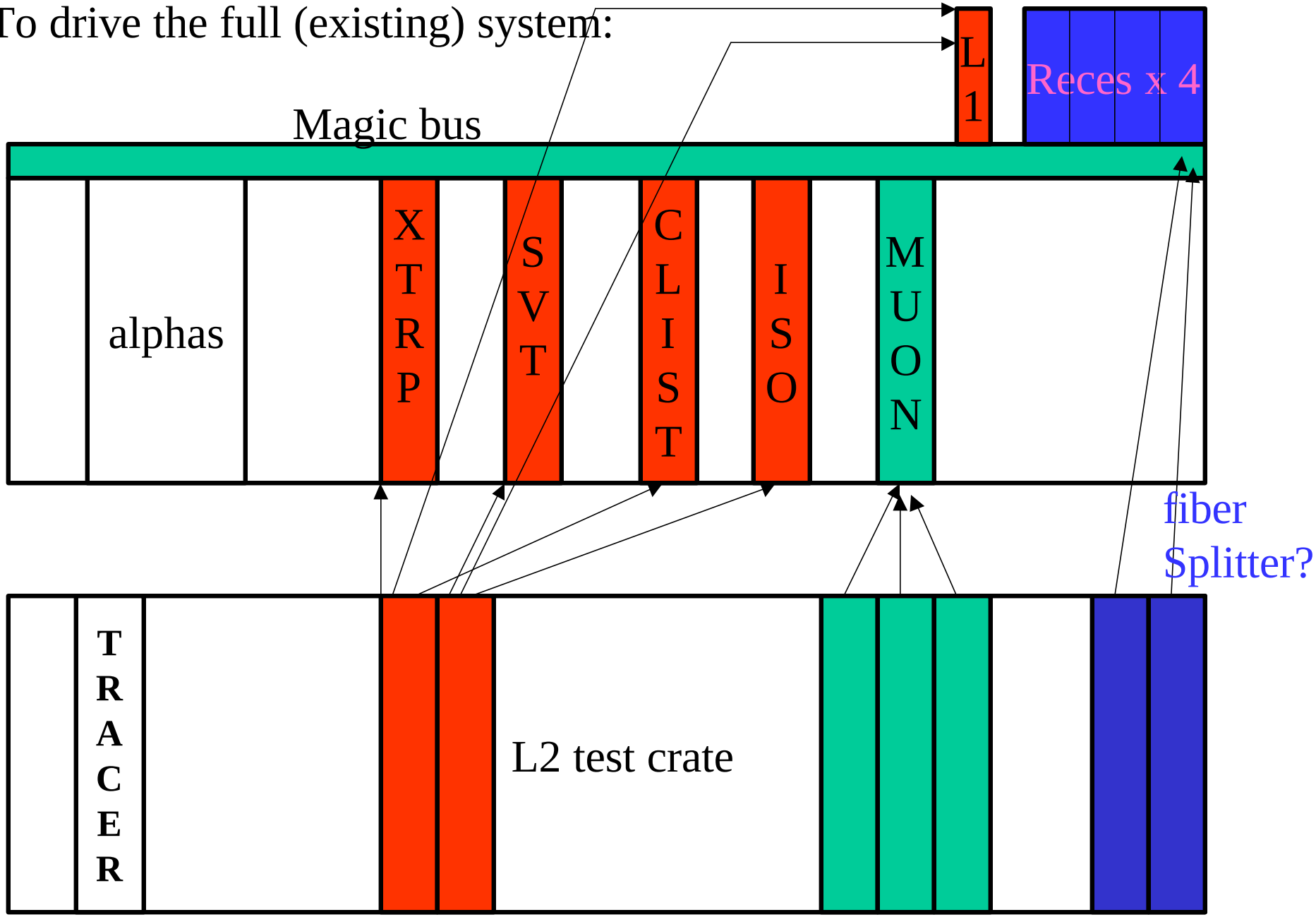
As source and sink

Board level test: Teststand Setup



Data source: universal test board
Data sink: either test board itself, or use Alpha if available for teststand

To drive the full (existing) system:



Summary of what the test board can do:

Board Level Testing:

- Test/debug each individual interface board (path);
- record real data sources (catch error conditions upstream etc);

System Level Testing:

Drive the full L2 decision crate, with different event patterns/size, different latency for each data path, with different L1A rates. Allow us to debug system level problems, optimize and evaluate overall system performance standalone (software & firmware).

Need to find all crucial interface components:
(many of them are obsolete already!)

CY7B923-JC: hotlink Tx chip
CY7B933-JC: hotlink Rx chip
CY7C441/3: FIFO for Tx
CY7C451/3-14JC: FIFO for Rx
AMP269051: Optical Tx
AMP269052: Optical Rx

AM7968-175JC: TAXI Tx chip
AM7969-175JC: TAXI Rx chip
HFBR-1414T: Optical Tx
HFBR-2416T: Optical Rx

Found many of them already, but still need to find AMP269051/2 !

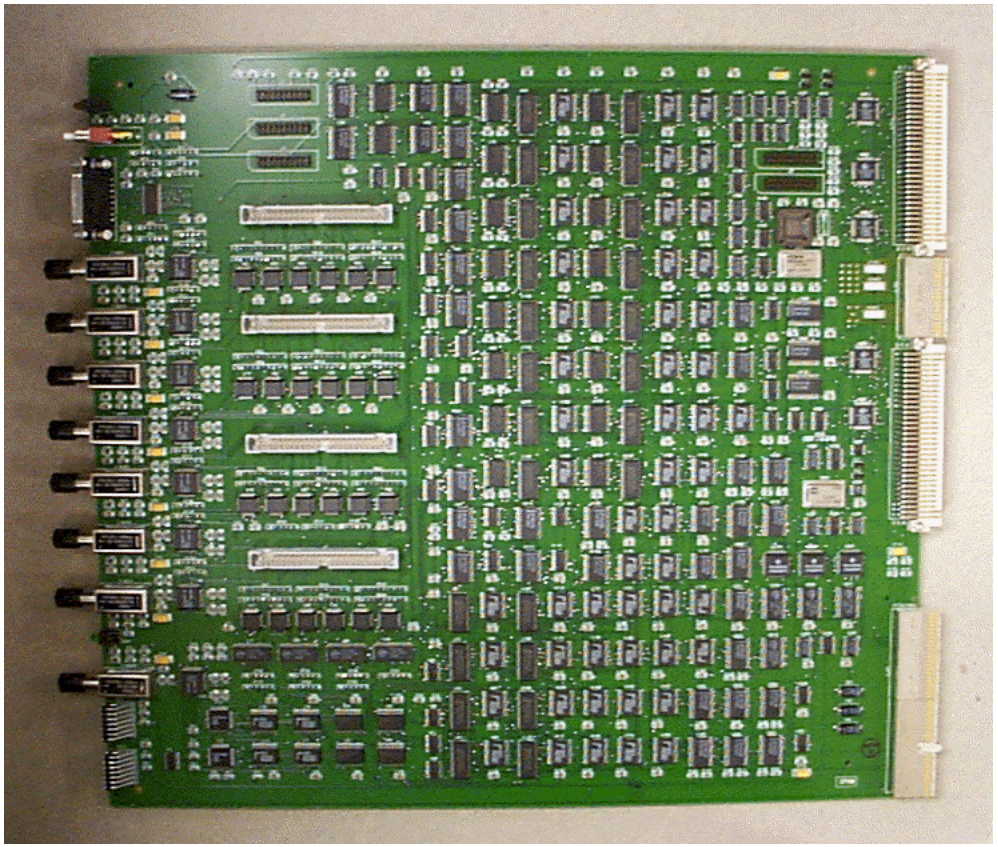
Will need support from all subsystem experts:

Need information on the design for the real
source and sink for each data path:

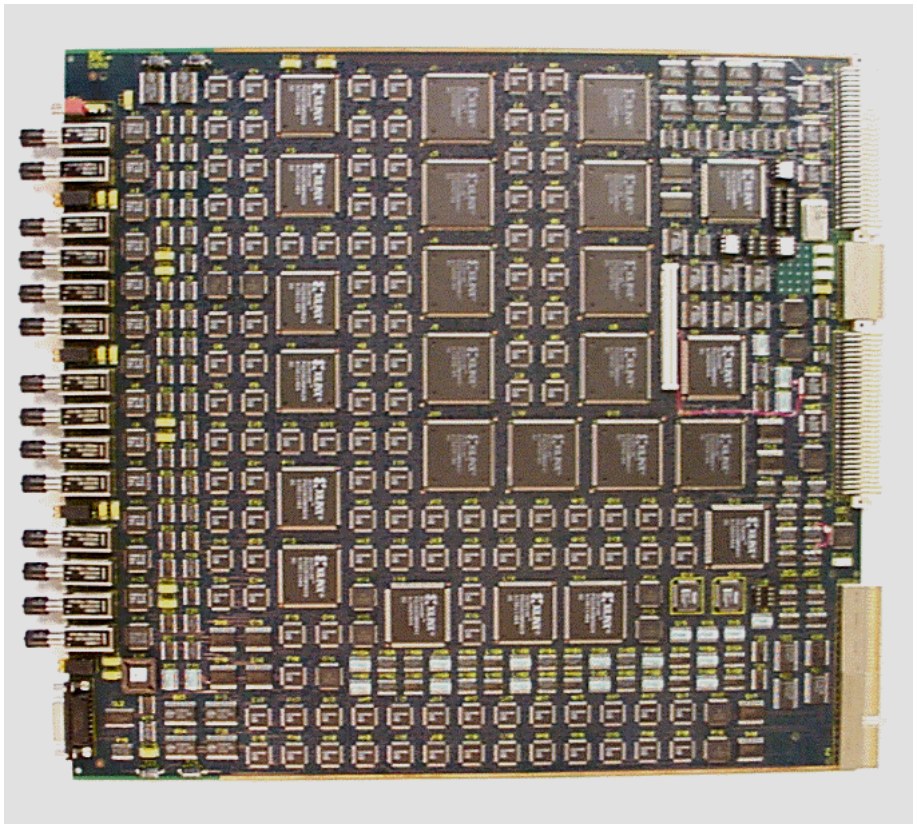
interface components, schematics, board layout for
the interface part, data format, response to Halt-Recover-Run
typical data size and range, typical latency and range…etc.

Documentation on each data path would really help

Examples: Hotlink fiber connections x8 and x14 per board

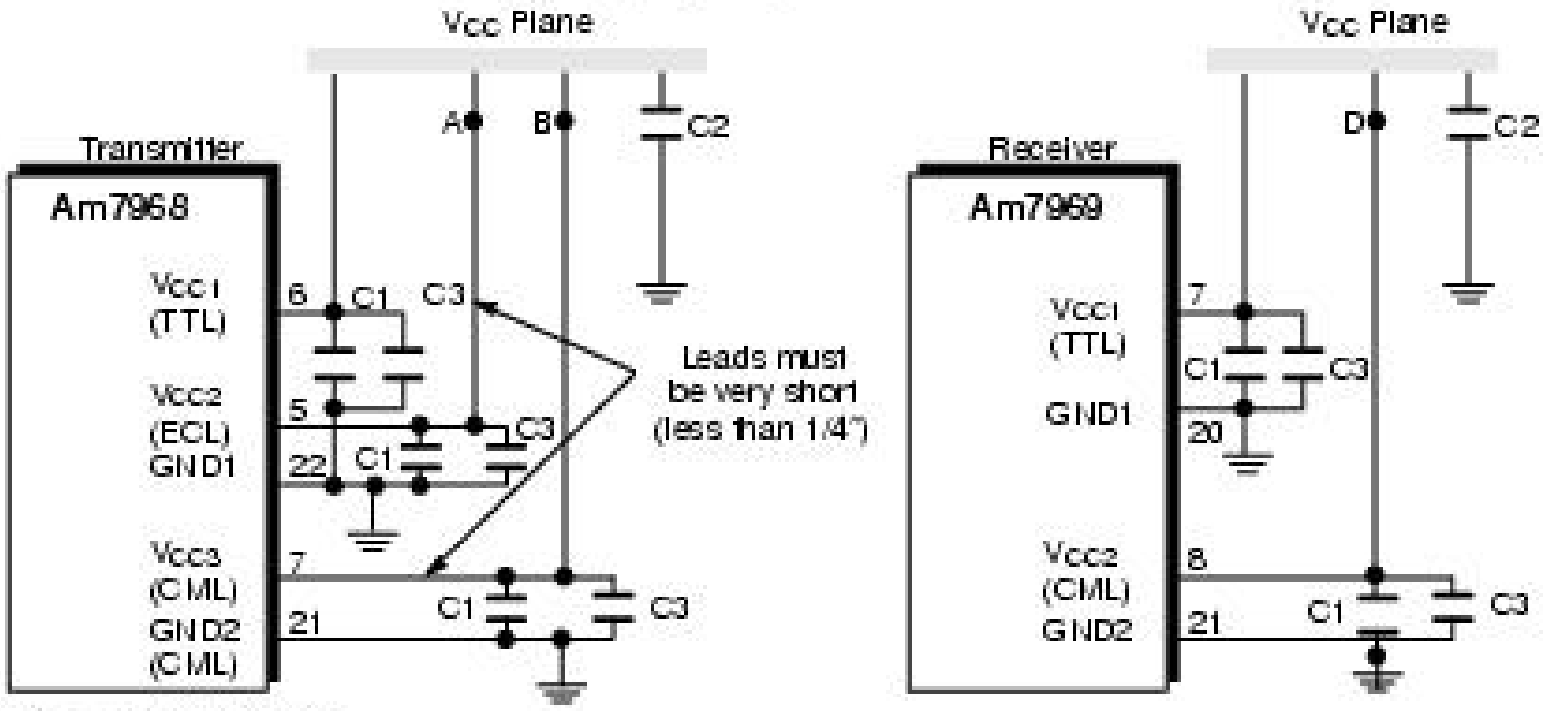


Muon Matchbox test card



Muon MatchBox

Figure 6-1 Transmitter and Receiver Decoupling Layouts

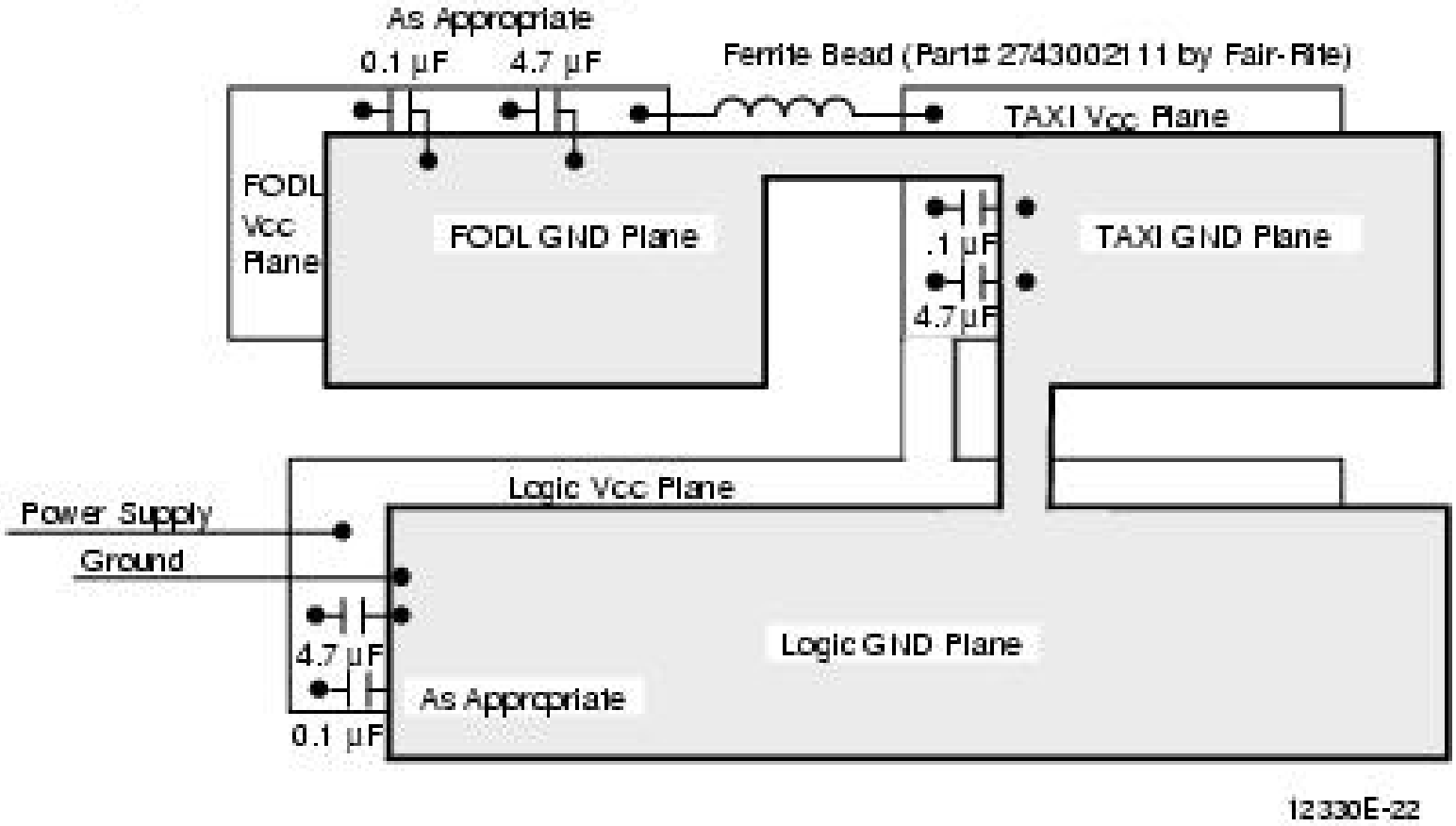


C1 = 0.1 µF (ceramic)
C2 = 1 µF Tantalum
C3 = 0.01 µF (ceramic)

12330E-19

To further decouple the TAXIchip set, it is highly recommended that ferrite beads be inserted at locations A, B and D.

Figure 6-5 Fiber Optic Data Link Decoupling



Note: This connection includes a ferrite bead in the V_{CC} circuit of the fiber optic components.